(12) UK Patent Application (19) GB (11) 2 351 216 (13) A

(43) Date of A Publication 20.12.2000

(21) Application No 0000088.5

(22) Date of Filing 05.01.2000

(30) Priority Data
(31) 9901230 (32) 20.01.1999 (33) GB
(31) 9921321 (32) 09.09.1999

(51) INT CL[7]
G06T 15/70

(52) UK CL (Edition R)
H4T TBAS

(56) Documents Cited
EP 0696018 A2 WO 99/30494 A1 WO 00/10099 A1
US 5491743 A

(58) Field of Search
UK CL (Edition R) H4T TBAS TBBA TBBD TBEX TCGD TCJA TCXX
INT CL[7] G06T 13/00 15/70 17/00
ONLINE: EPOQUE

(71) Applicant(s)
Canon Kabushiki Kaisha
(Incorporated in Japan)
3-30-2 Shimomaruko, Ohta-Ku, Tokyo 146, Japan

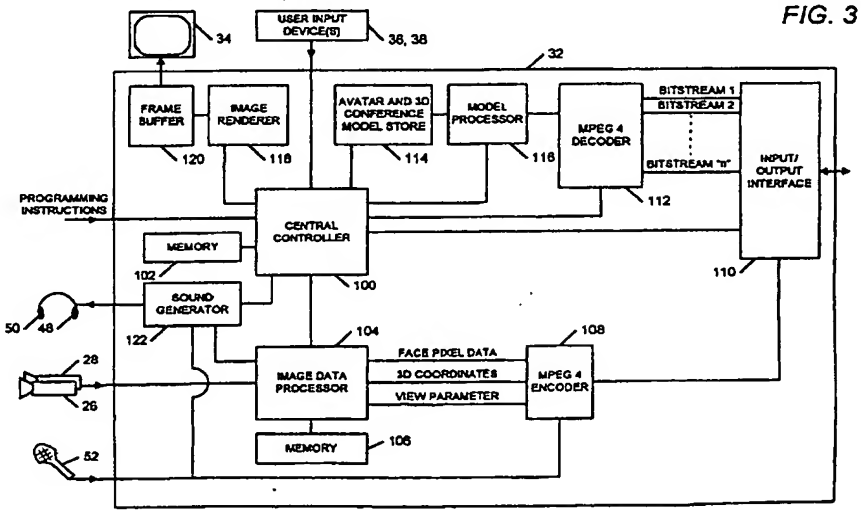
(72) Inventor(s)
Michael James Taylor
Simon Michael Rowe
Charles Stephen Wiles

(74) Agent and/or Address for Service
Beresford & Co
2-5 Warwick Court, High Holborn, LONDON,
WC1R 5DJ, United Kingdom

(54) Abstract Title
Computer conferencing apparatus

(57) In a desktop video conferencing system, a plurality of user stations 2-14 are interconnected to exchange data. Each user station 2-14 stores data defining a respective, different three-dimensional computer model containing avatars of the participants in the conference, and displays images of the model to a user. Each user station uses a pair of cameras 26, 28 to record images of the user wearing coloured body markers 70, 72 and a headset 30 on which lights 56-64 are mounted. The image data is processed to determine the movements of the user and the point in the displayed images at which the user is looking. This information is transmitted to the other user stations where it is used to animate the corresponding avatar. Each avatar is animated so that movements of its head correspond to scaled movements of the user's head. Because the computer model at each user station is different, the movement of the head of a given avatar is different at each user station. Image data is processed in each user station to determine the type of transformation which relates the positions of the camera 26, 28 to be used in determining the user's movements. The sound of the other participants is output to the headset 30 worn by a user. The sound is adjusted in dependence upon the position of the user's head.

FIG. 3



BEST AVAILABLE COPY

GB 2 351 216 A

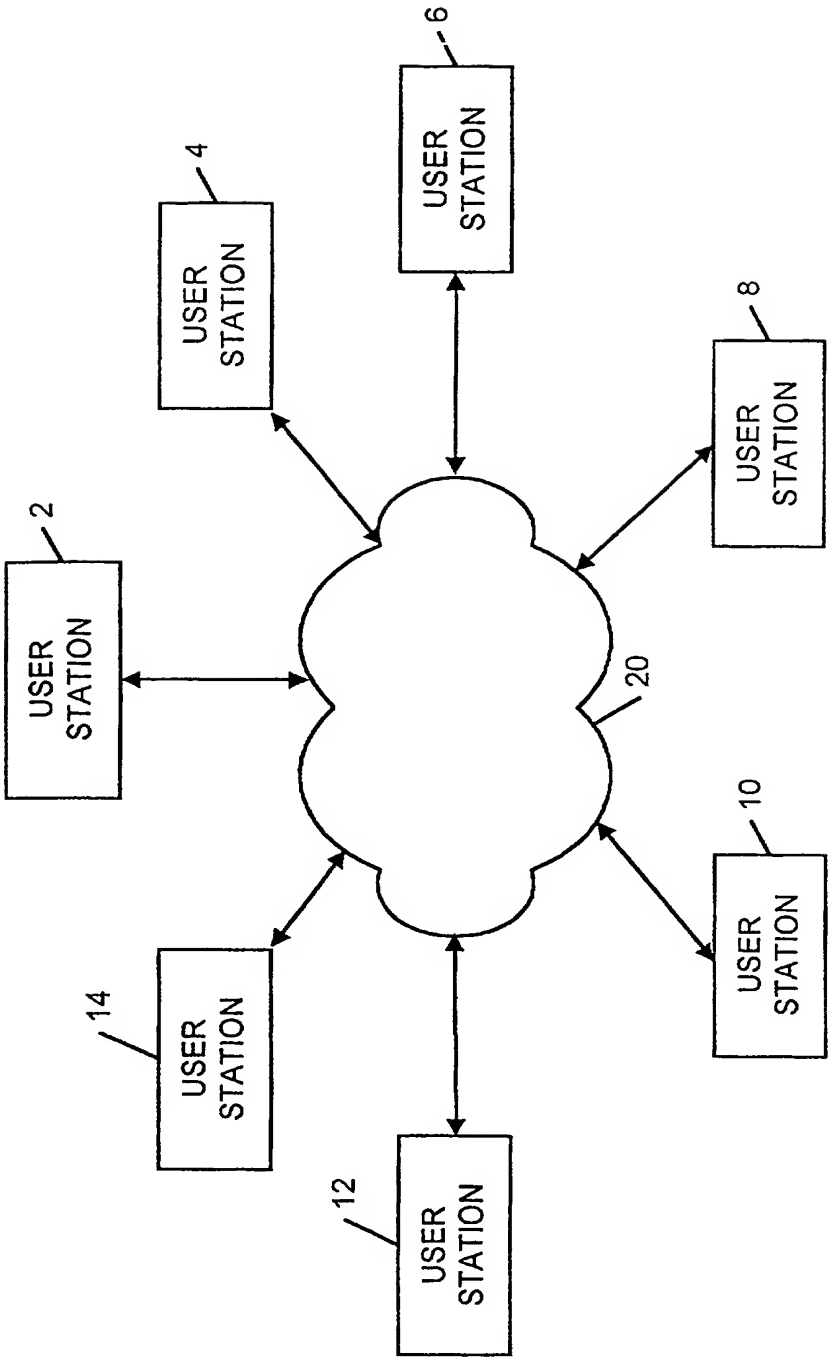

USER STATION
2
USER STATION
4
USER STATION
6
USER STATION
8
USER STATION
10
USER STATION
12
USER STATION
14
20

FIG. 1

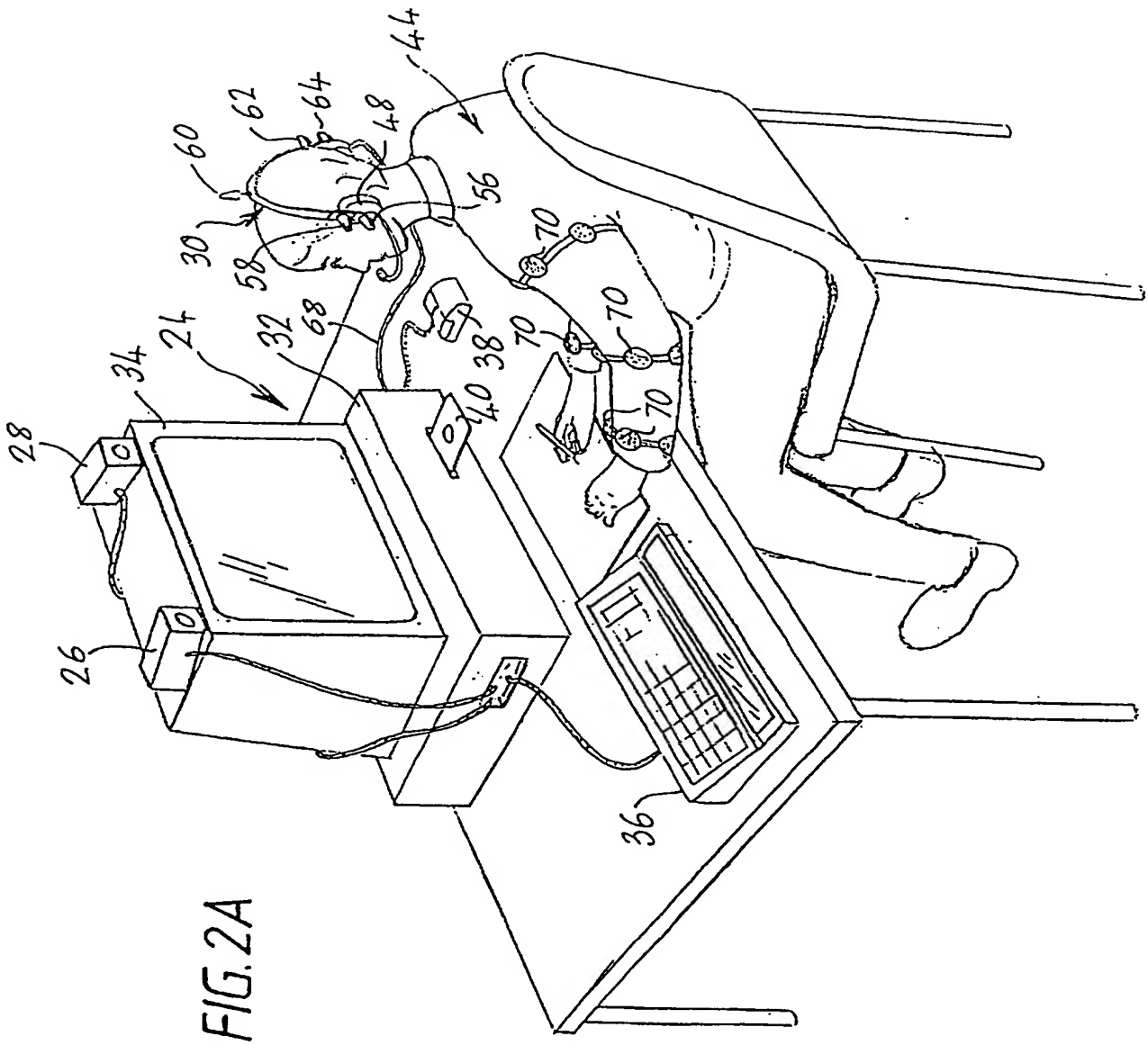


FIG. 2A

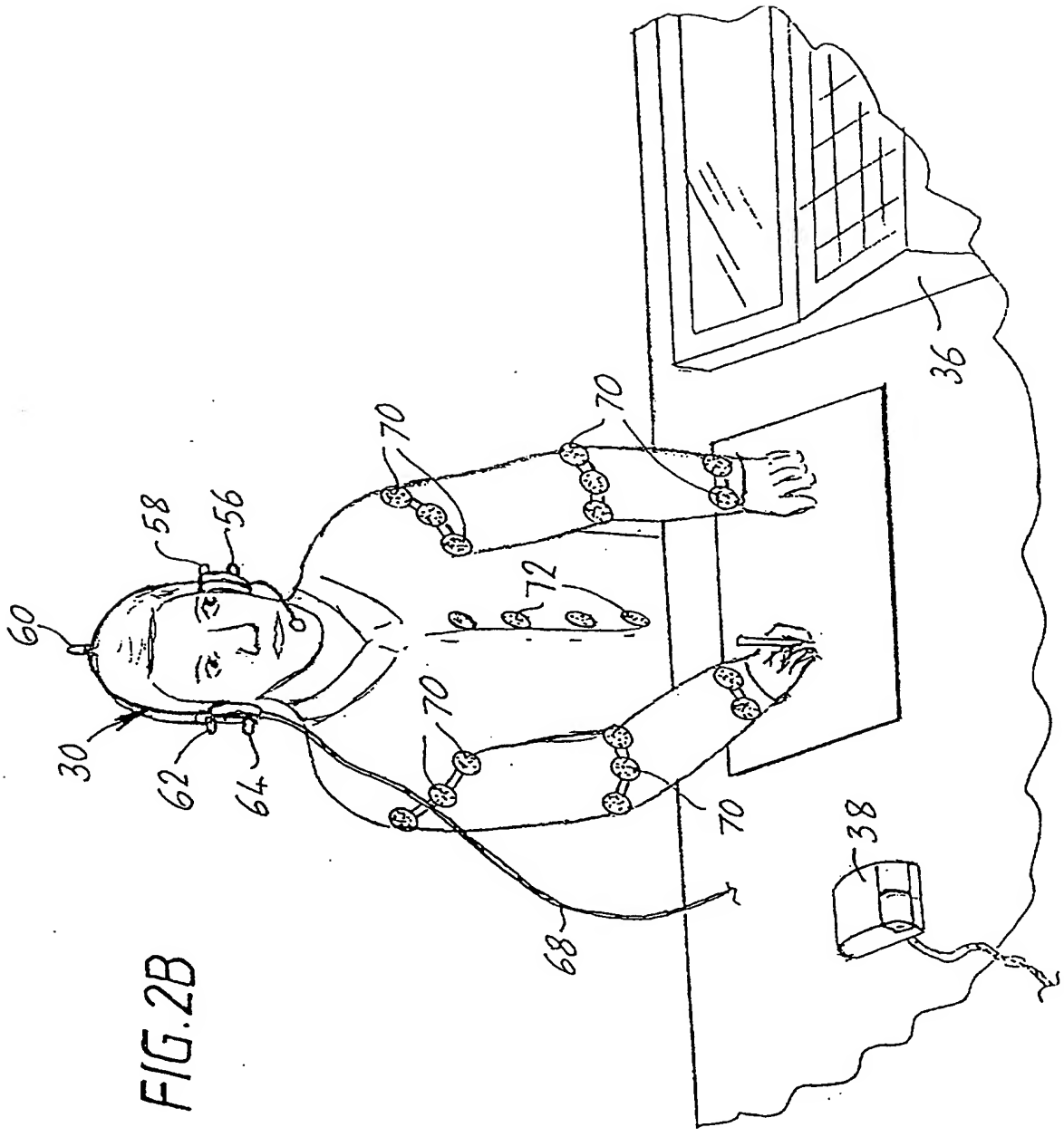


FIG. 2B

FIG. 2C

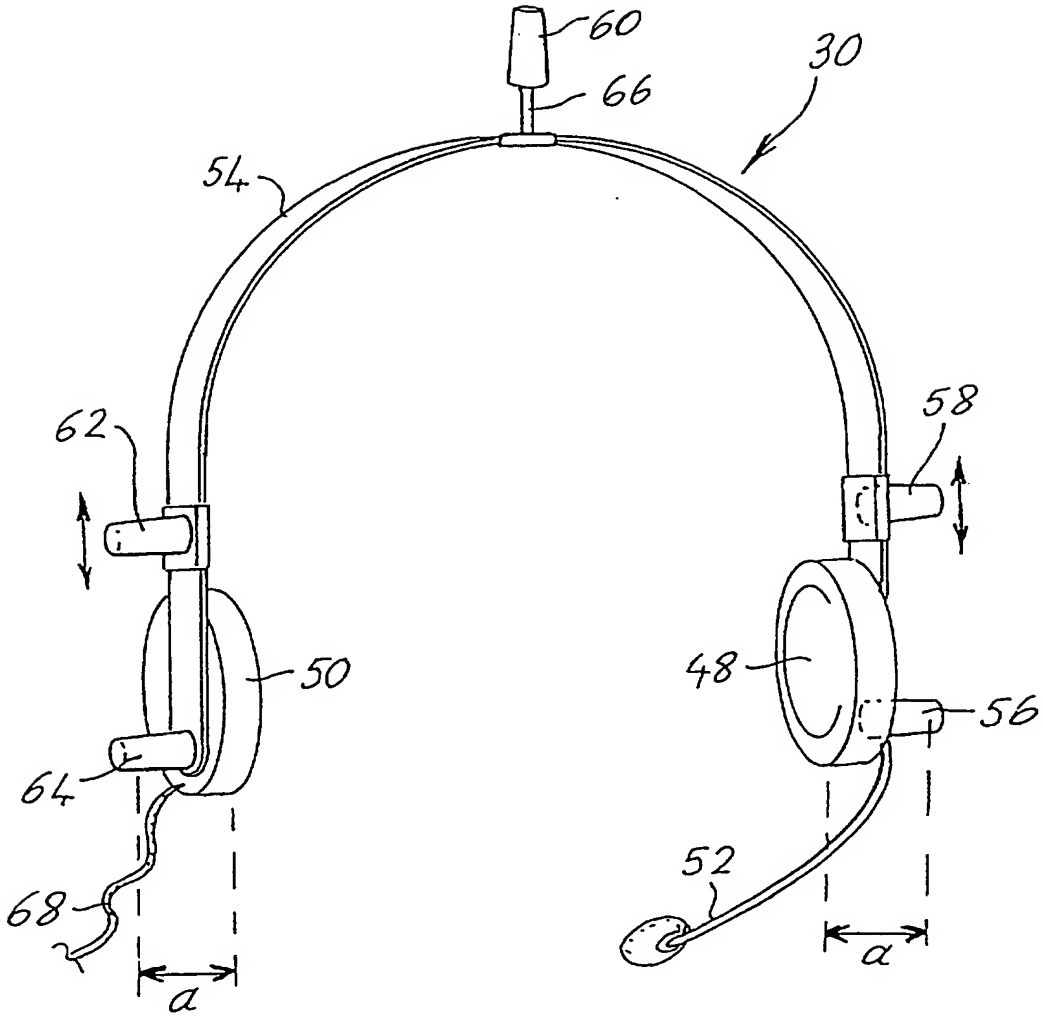
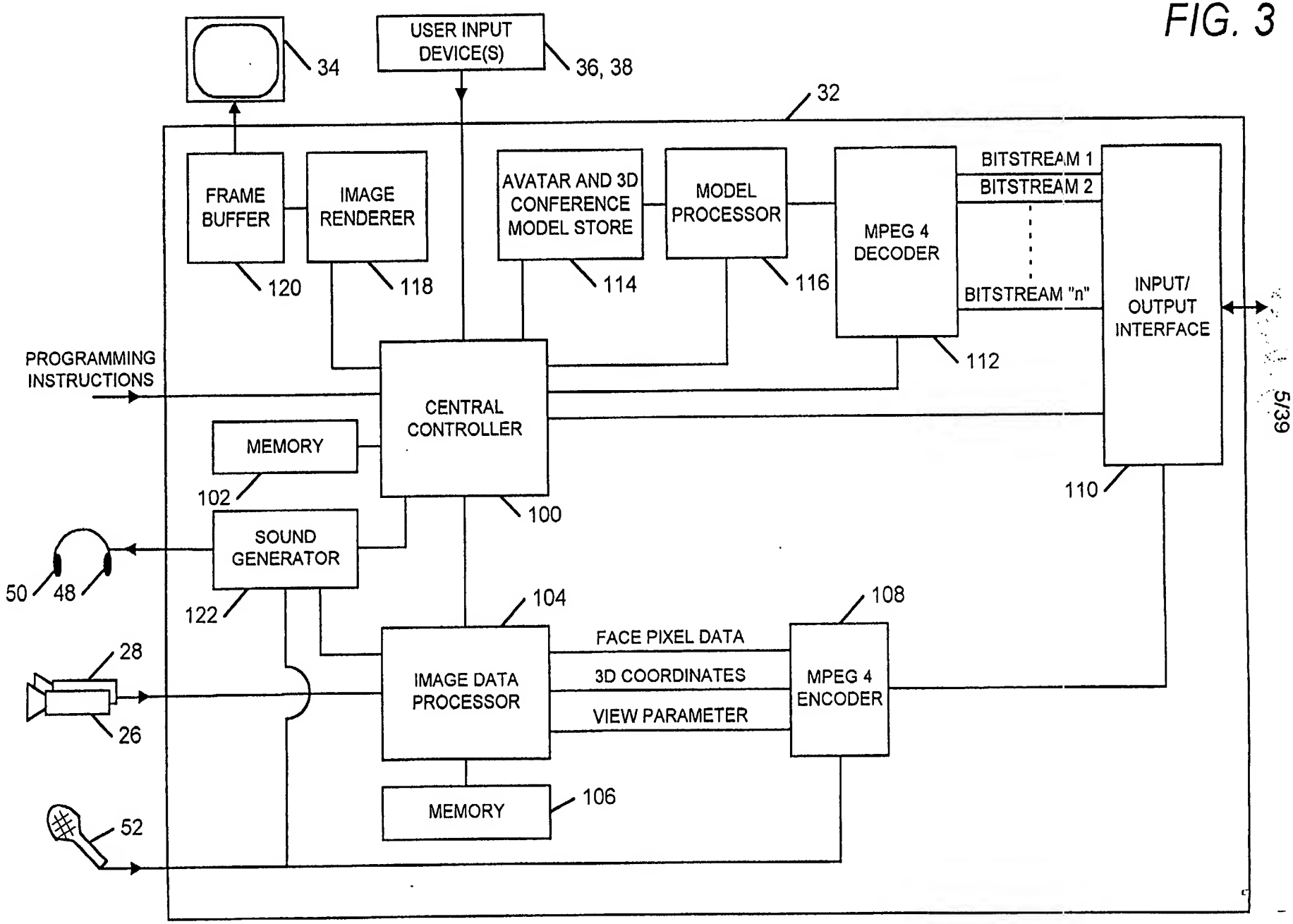


FIG. 3


34
USER INPUT DEVICE(S)
36, 38
32
FRAME BUFFER
IMAGE RENDERER
AVATAR AND 3D CONFERENCE MODEL STORE
MODEL PROCESSOR
MPEG 4 DECODER
BITSTREAM 1
BITSTREAM 2
BITSTREAM "n"
INPUT/ OUTPUT INTERFACE
120
118
114
116
112
PROGRAMMING INSTRUCTIONS
CENTRAL CONTROLLER
MEMORY
102
100
110
SOUND GENERATOR
50
48
122
104
108
28
IMAGE DATA PROCESSOR
FACE PIXEL DATA
3D COORDINATES
VIEW PARAMETER
MPEG 4 ENCODER
26
MEMORY
106
52

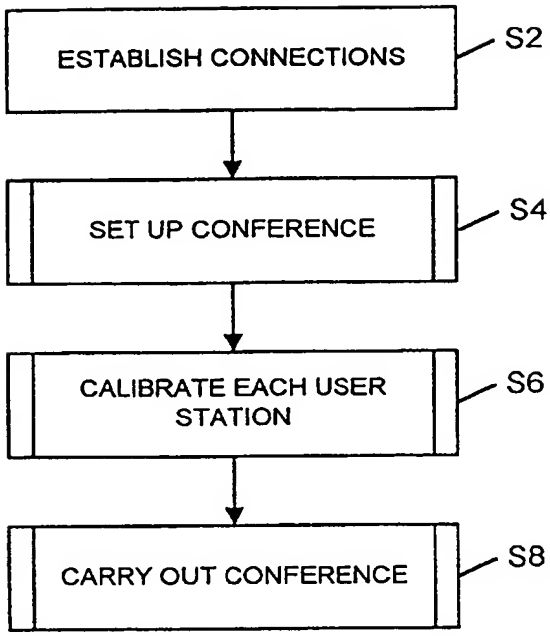

ESTABLISH CONNECTIONS
S2
SET UP CONFERENCE
S4
CALIBRATE EACH USER STATION
S6
CARRY OUT CONFERENCE
S8

FIG. 4

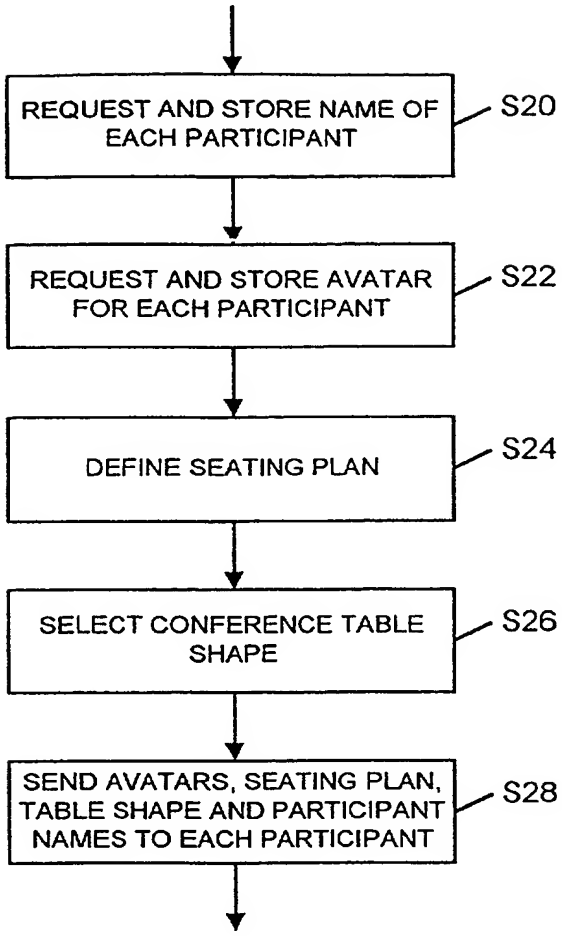

REQUEST AND STORE NAME OF EACH PARTICIPANT
S20
REQUEST AND STORE AVATAR FOR EACH PARTICIPANT
S22
DEFINE SEATING PLAN
S24
SELECT CONFERENCE TABLE SHAPE
S26
SEND AVATARS, SEATING PLAN, TABLE SHAPE AND PARTICIPANT NAMES TO EACH PARTICIPANT
S28

FIG. 5

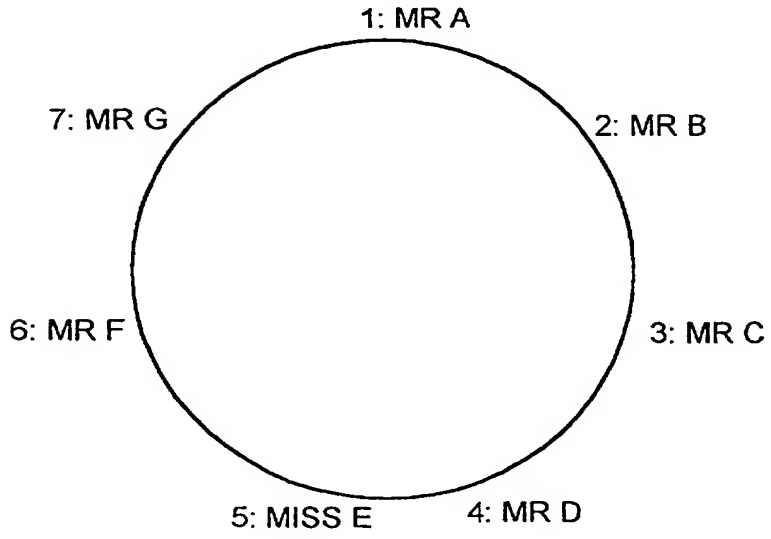

1: MR A
2: MR B
3: MR C
4: MR D
5: MISS E
6: MR F
7: MR G

FIG. 6

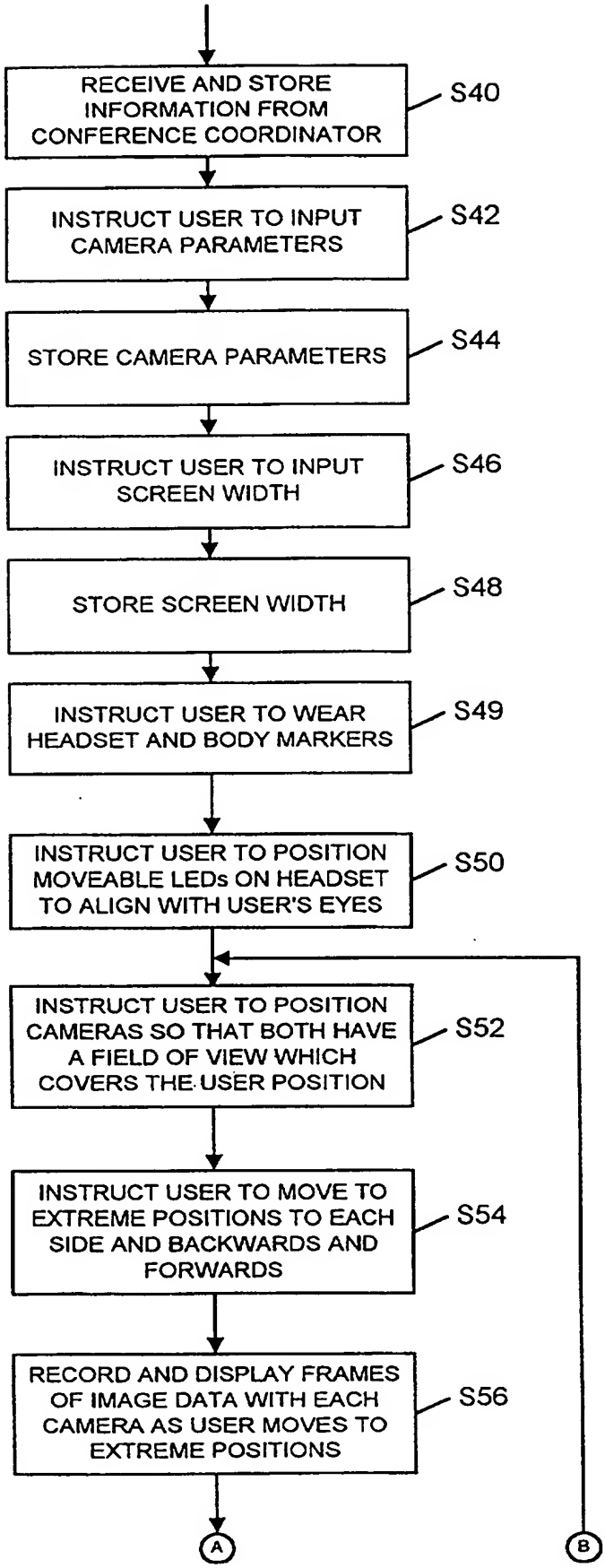

RECEIVE AND STORE INFORMATION FROM CONFERENCE COORDINATOR
S40
INSTRUCT USER TO INPUT CAMERA PARAMETERS
S42
STORE CAMERA PARAMETERS
S44
INSTRUCT USER TO INPUT SCREEN WIDTH
S46
STORE SCREEN WIDTH
S48
INSTRUCT USER TO WEAR HEADSET AND BODY MARKERS
S49
INSTRUCT USER TO POSITION MOVEABLE LEDs ON HEADSET TO ALIGN WITH USER'S EYES
S50
INSTRUCT USER TO POSITION CAMERAS SO THAT BOTH HAVE A FIELD OF VIEW WHICH COVERS THE USER POSITION
S52
INSTRUCT USER TO MOVE TO EXTREME POSITIONS TO EACH SIDE AND BACKWARDS AND FORWARDS
S54
RECORD AND DISPLAY FRAMES OF IMAGE DATA WITH EACH CAMERA AS USER MOVES TO EXTREME POSITIONS
S56
A
B

FIG. 7

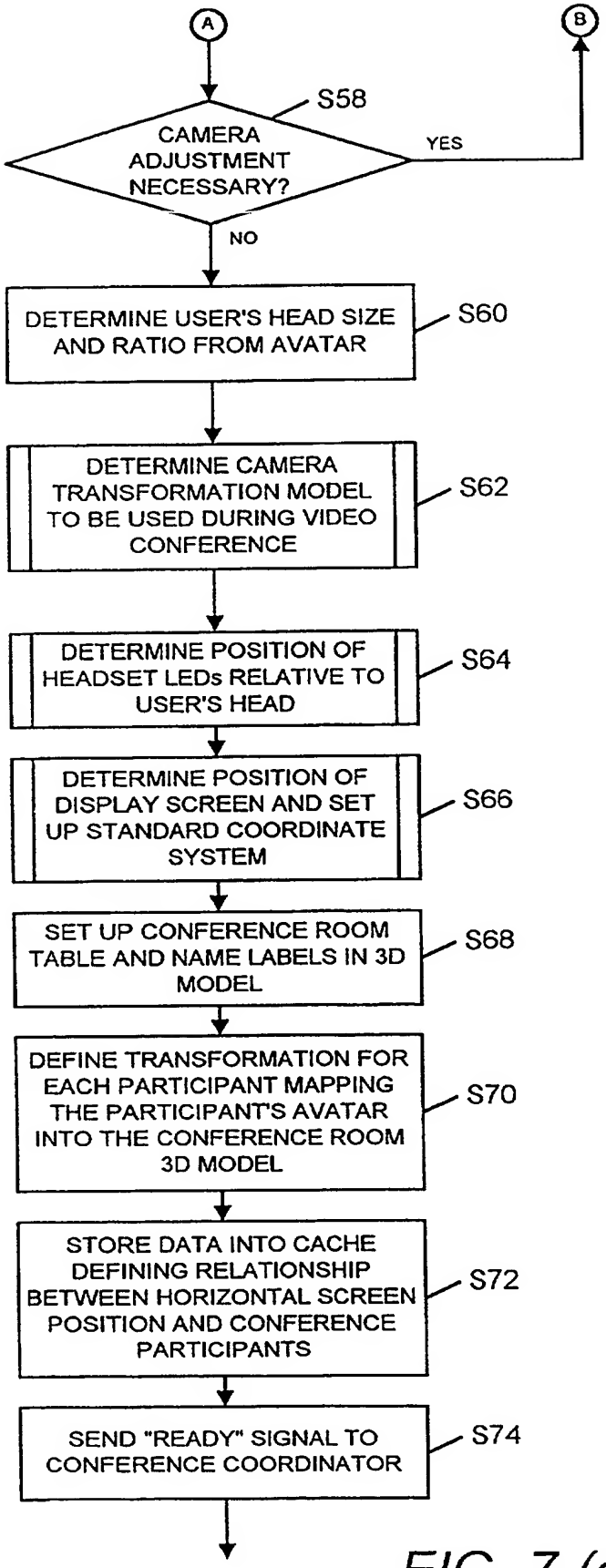

A
B
S58
CAMERA ADJUSTMENT NECESSARY?
YES
NO
DETERMINE USER'S HEAD SIZE AND RATIO FROM AVATAR
S60
DETERMINE CAMERA TRANSFORMATION MODEL TO BE USED DURING VIDEO CONFERENCE
S62
DETERMINE POSITION OF HEADSET LEDs RELATIVE TO USER'S HEAD
S64
DETERMINE POSITION OF DISPLAY SCREEN AND SET UP STANDARD COORDINATE SYSTEM
S66
SET UP CONFERENCE ROOM TABLE AND NAME LABELS IN 3D MODEL
S68
DEFINE TRANSFORMATION FOR EACH PARTICIPANT MAPPING THE PARTICIPANT'S AVATAR INTO THE CONFERENCE ROOM 3D MODEL
S70
STORE DATA INTO CACHE DEFINING RELATIONSHIP BETWEEN HORIZONTAL SCREEN POSITION AND CONFERENCE PARTICIPANTS
S72
SEND "READY" SIGNAL TO CONFERENCE COORDINATOR
S74

FIG. 7 (cont)

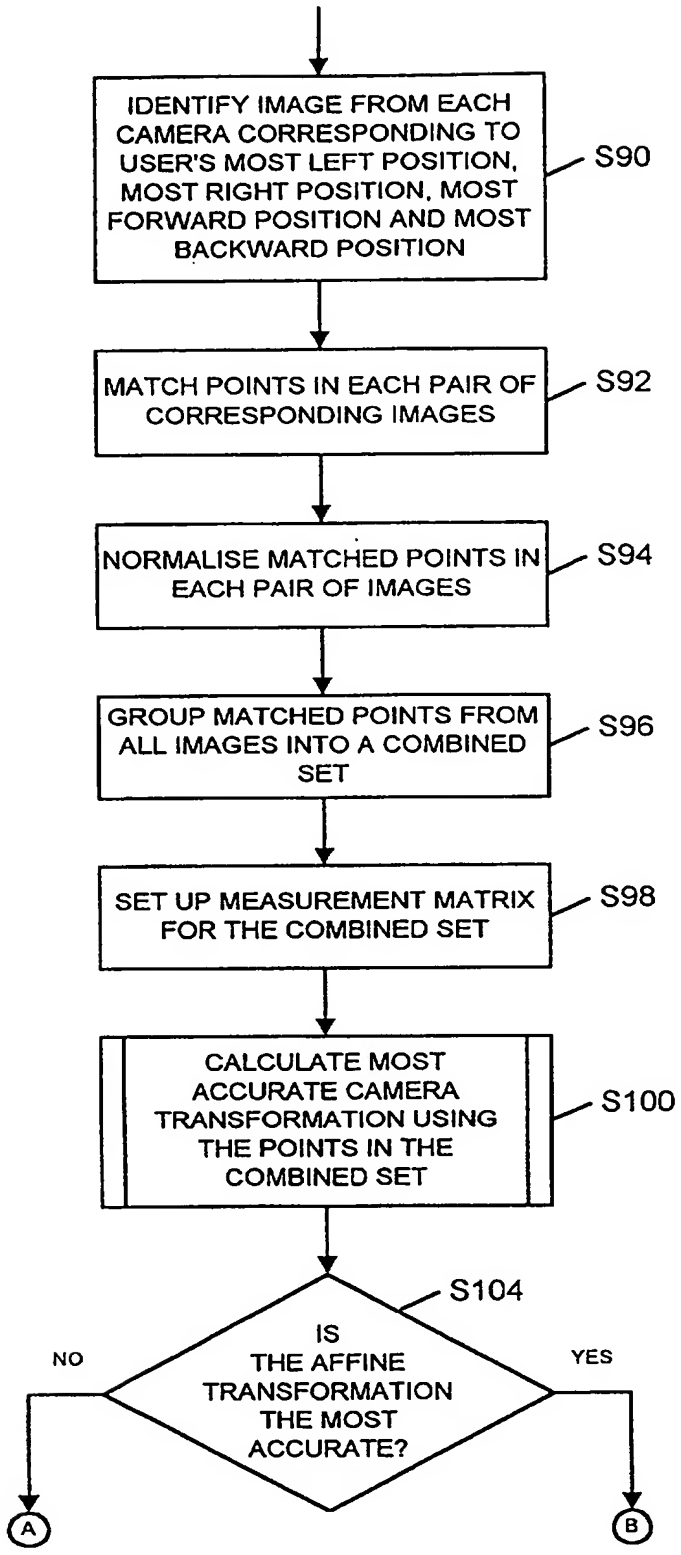

IDENTIFY IMAGE FROM EACH CAMERA CORRESPONDING TO USER'S MOST LEFT POSITION, MOST RIGHT POSITION, MOST FORWARD POSITION AND MOST BACKWARD POSITION
S90
MATCH POINTS IN EACH PAIR OF CORRESPONDING IMAGES
S92
NORMALISE MATCHED POINTS IN EACH PAIR OF IMAGES
S94
GROUP MATCHED POINTS FROM ALL IMAGES INTO A COMBINED SET
S96
SET UP MEASUREMENT MATRIX FOR THE COMBINED SET
S98
CALCULATE MOST ACCURATE CAMERA TRANSFORMATION USING THE POINTS IN THE COMBINED SET
S100
S104
IS THE AFFINE TRANSFORMATION THE MOST ACCURATE?
NO
YES
A
B

FIG. 8

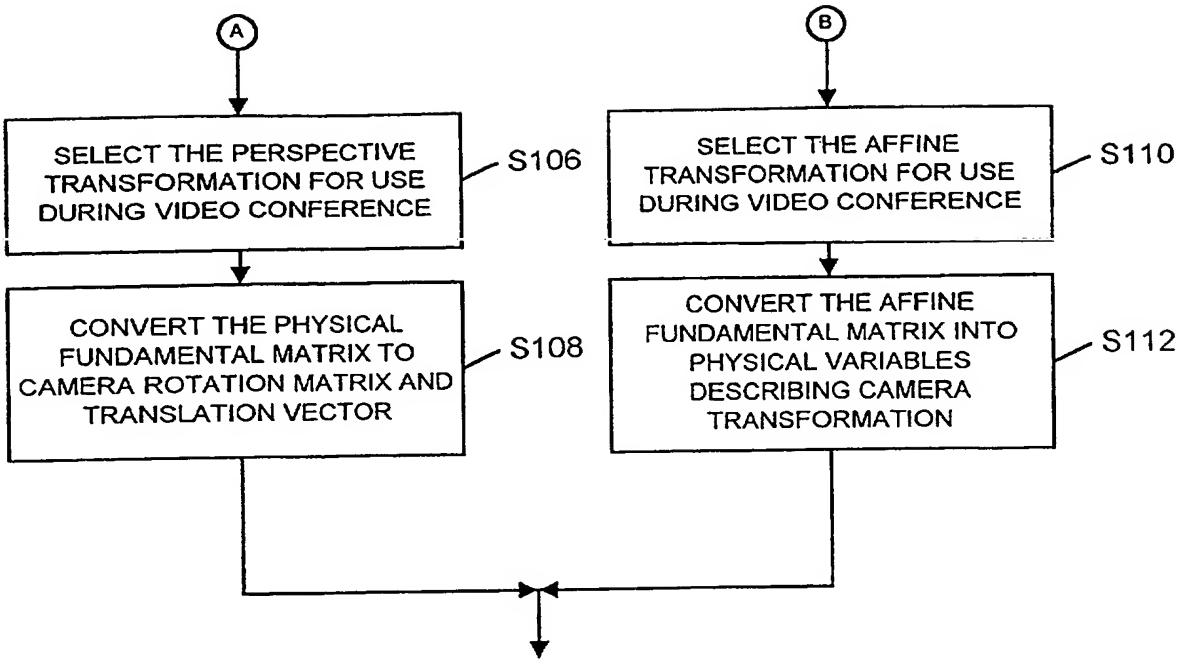


FIG. 8 (cont)

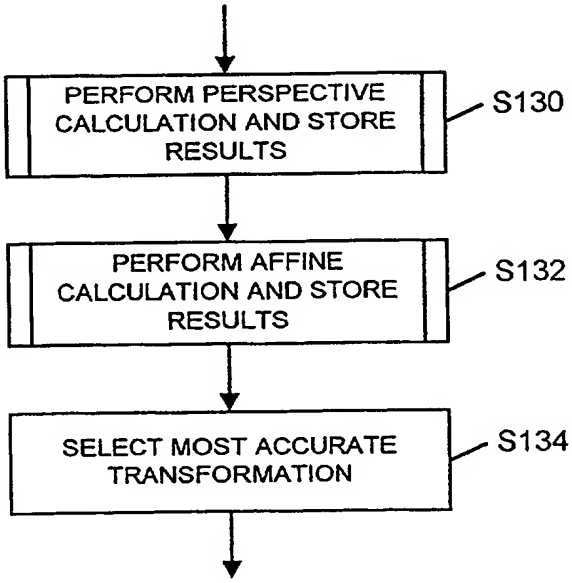


FIG. 9

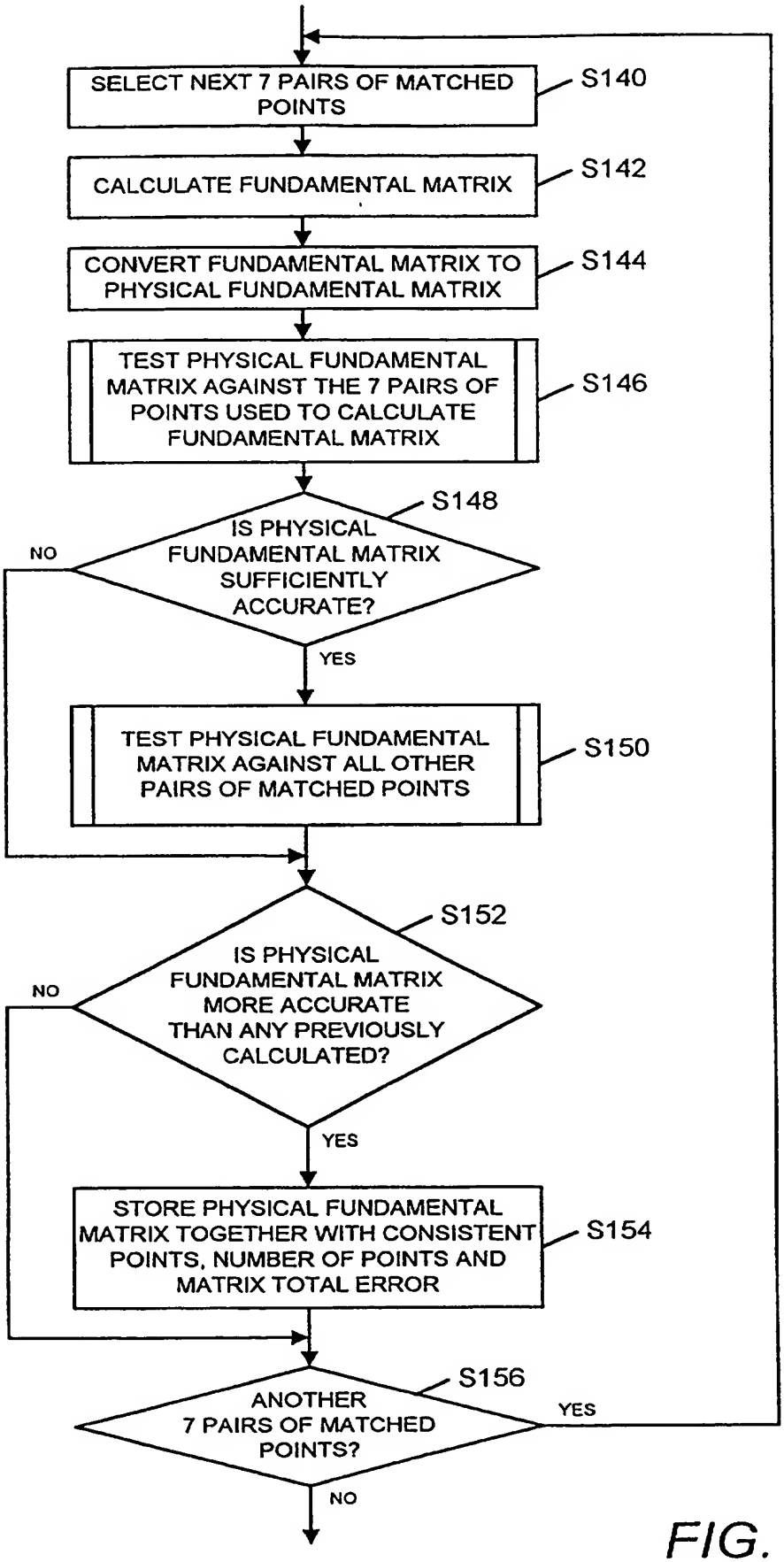

SELECT NEXT 7 PAIRS OF MATCHED POINTS
S140
CALCULATE FUNDAMENTAL MATRIX
S142
CONVERT FUNDAMENTAL MATRIX TO PHYSICAL FUNDAMENTAL MATRIX
S144
TEST PHYSICAL FUNDAMENTAL MATRIX AGAINST THE 7 PAIRS OF POINTS USED TO CALCULATE FUNDAMENTAL MATRIX
S146
S148
IS PHYSICAL FUNDAMENTAL MATRIX SUFFICIENTLY ACCURATE?
NO
YES
TEST PHYSICAL FUNDAMENTAL MATRIX AGAINST ALL OTHER PAIRS OF MATCHED POINTS
S150
S152
IS PHYSICAL FUNDAMENTAL MATRIX MORE ACCURATE THAN ANY PREVIOUSLY CALCULATED?
NO
YES
STORE PHYSICAL FUNDAMENTAL MATRIX TOGETHER WITH CONSISTENT POINTS, NUMBER OF POINTS AND MATRIX TOTAL ERROR
S154
S156
ANOTHER 7 PAIRS OF MATCHED POINTS?
YES
NO

FIG. 10

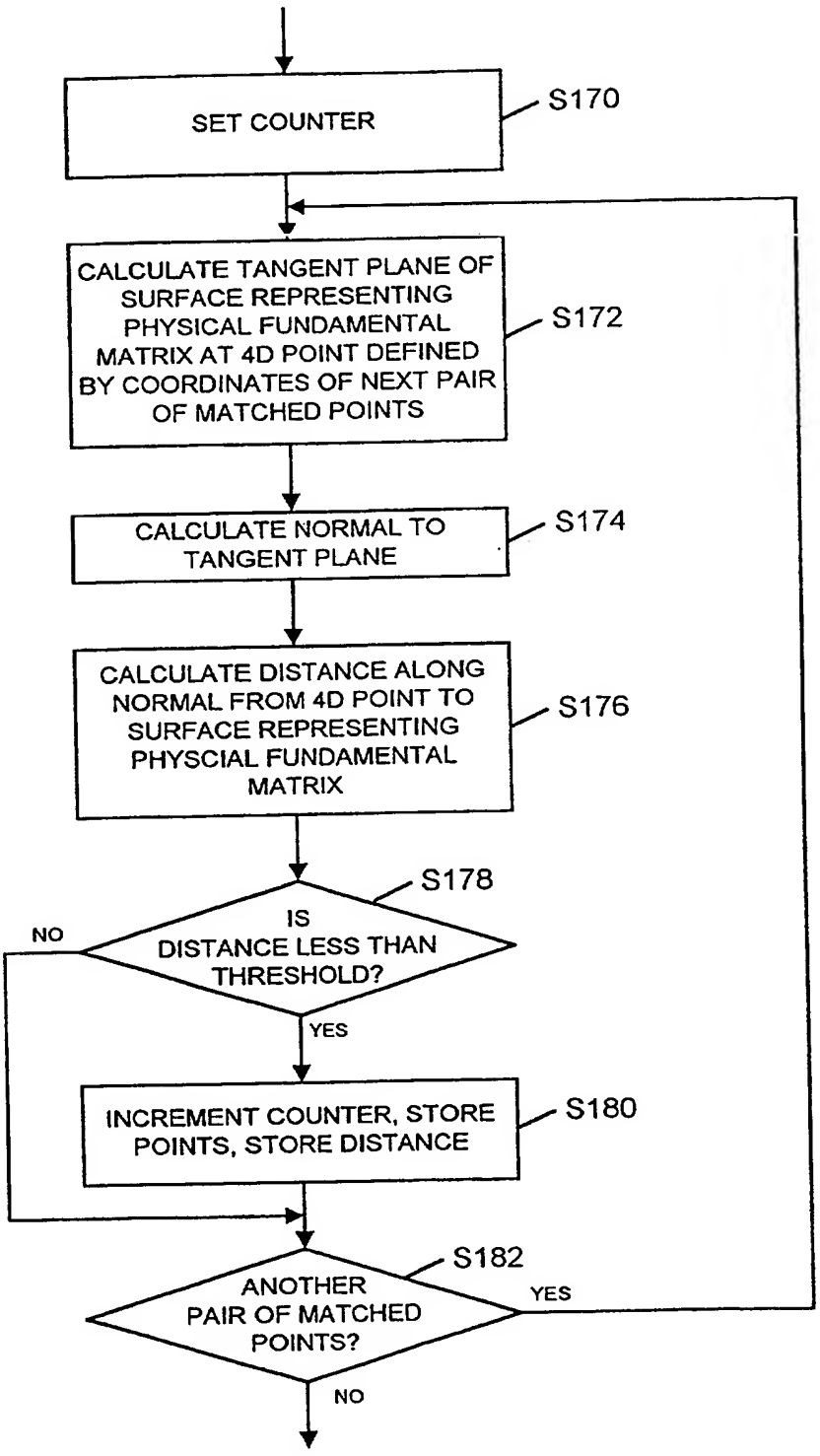

SET COUNTER
S170
CALCULATE TANGENT PLANE OF SURFACE REPRESENTING PHYSICAL FUNDAMENTAL MATRIX AT 4D POINT DEFINED BY COORDINATES OF NEXT PAIR OF MATCHED POINTS
S172
CALCULATE NORMAL TO TANGENT PLANE
S174
CALCULATE DISTANCE ALONG NORMAL FROM 4D POINT TO SURFACE REPRESENTING PHYSCIAL FUNDAMENTAL MATRIX
S176
S178
IS DISTANCE LESS THAN THRESHOLD?
NO
YES
INCREMENT COUNTER, STORE POINTS, STORE DISTANCE
S180
S182
ANOTHER PAIR OF MATCHED POINTS?
YES
NO

FIG. 11

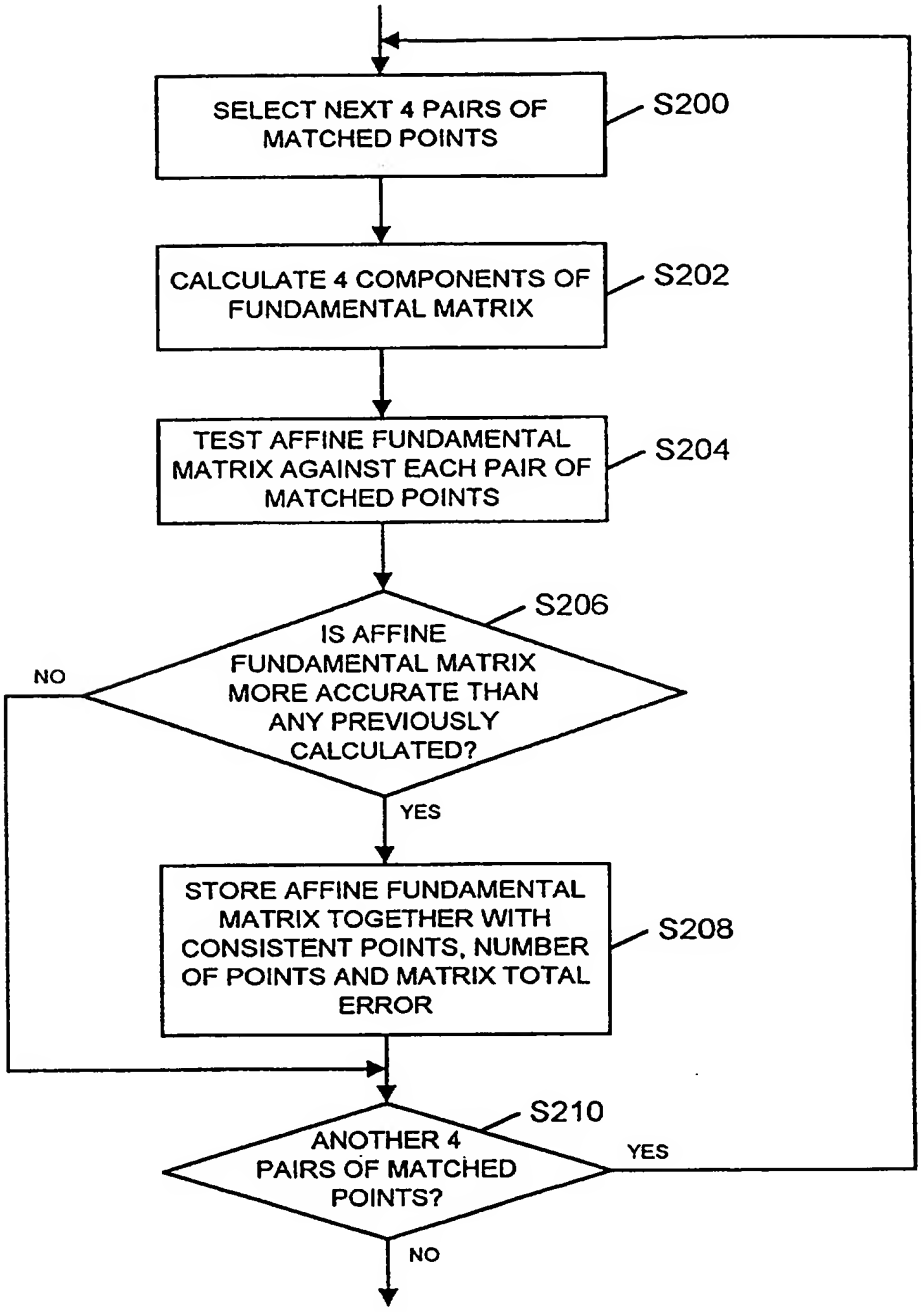

SELECT NEXT 4 PAIRS OF MATCHED POINTS
S200
CALCULATE 4 COMPONENTS OF FUNDAMENTAL MATRIX
S202
TEST AFFINE FUNDAMENTAL MATRIX AGAINST EACH PAIR OF MATCHED POINTS
S204
S206
IS AFFINE FUNDAMENTAL MATRIX MORE ACCURATE THAN ANY PREVIOUSLY CALCULATED?
NO
YES
STORE AFFINE FUNDAMENTAL MATRIX TOGETHER WITH CONSISTENT POINTS, NUMBER OF POINTS AND MATRIX TOTAL ERROR
S208
S210
ANOTHER 4 PAIRS OF MATCHED POINTS?
YES
NO

FIG. 12

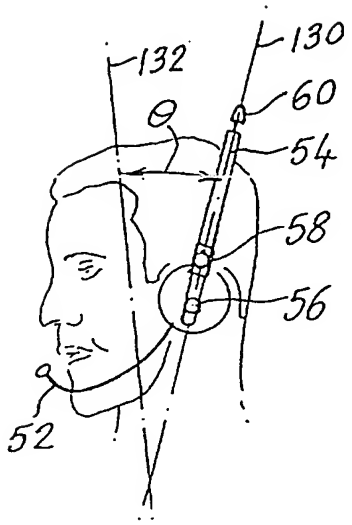

132
130
60
54
58
56
52

FIG. 13

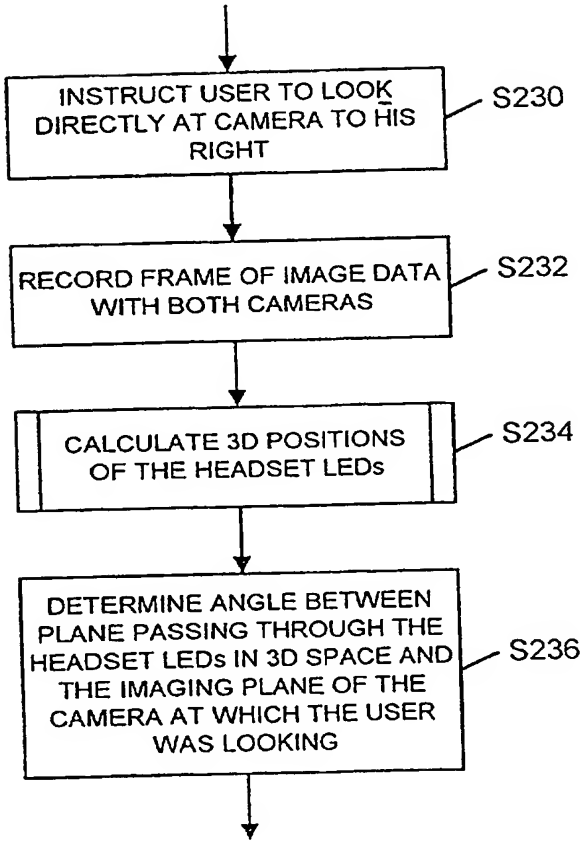

INSTRUCT USER TO LOOK DIRECTLY AT CAMERA TO HIS RIGHT
S230
RECORD FRAME OF IMAGE DATA WITH BOTH CAMERAS
S232
CALCULATE 3D POSITIONS OF THE HEADSET LEDs
S234
DETERMINE ANGLE BETWEEN PLANE PASSING THROUGH THE HEADSET LEDs IN 3D SPACE AND THE IMAGING PLANE OF THE CAMERA AT WHICH THE USER WAS LOOKING
S236

FIG. 14

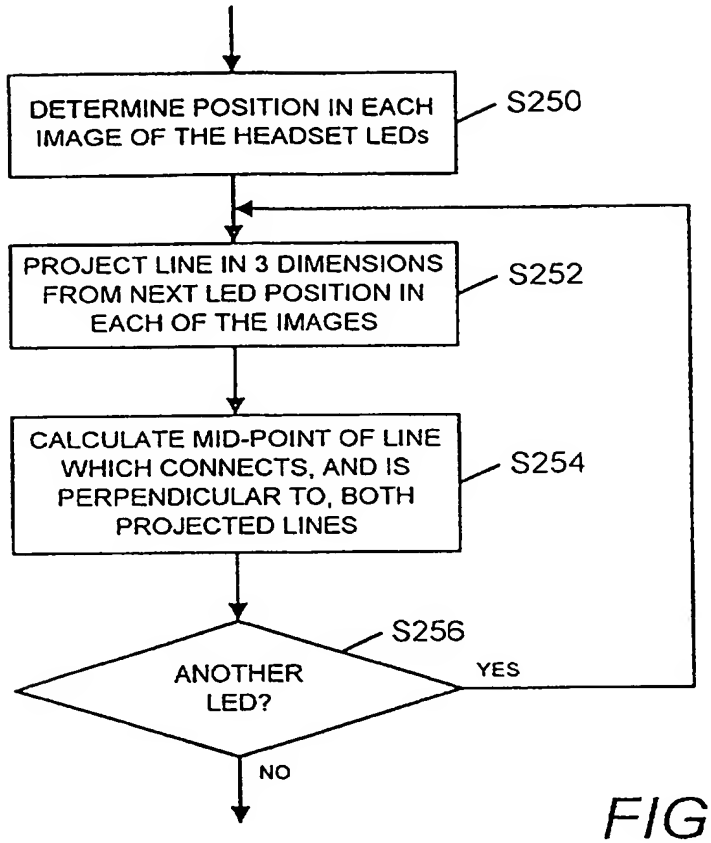

DETERMINE POSITION IN EACH IMAGE OF THE HEADSET LEDs
S250
PROJECT LINE IN 3 DIMENSIONS FROM NEXT LED POSITION IN EACH OF THE IMAGES
S252
CALCULATE MID-POINT OF LINE WHICH CONNECTS, AND IS PERPENDICULAR TO, BOTH PROJECTED LINES
S254
S256
ANOTHER LED?
YES
NO

FIG. 15

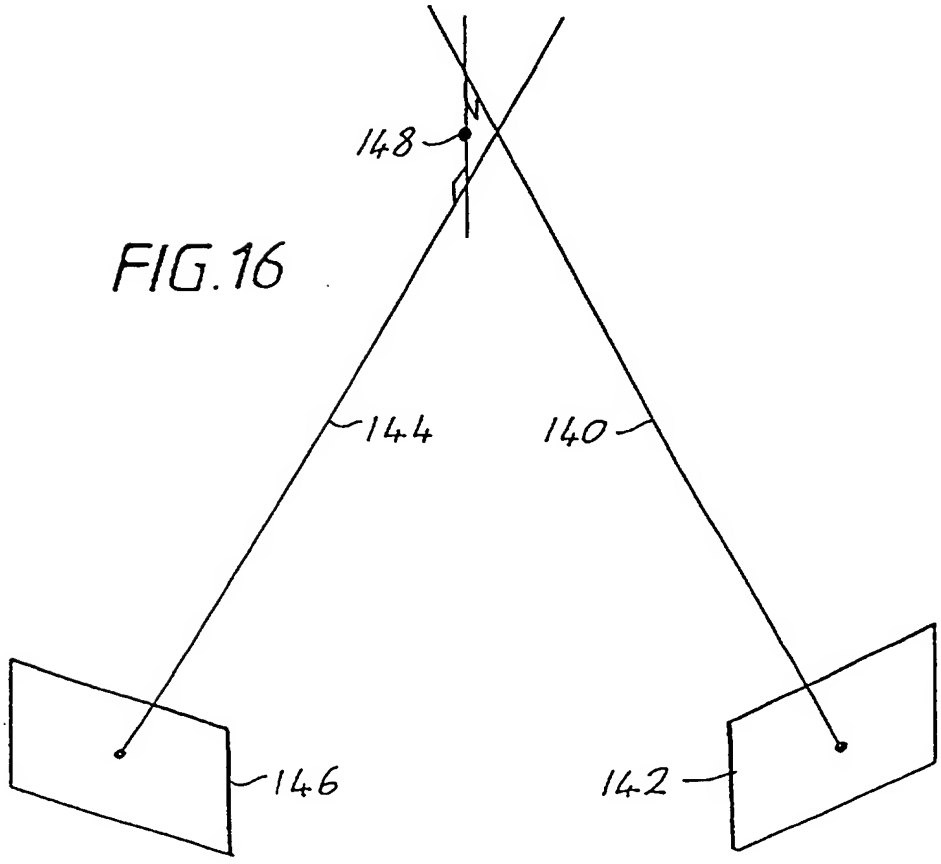

148
144
140
146
142

FIG.16

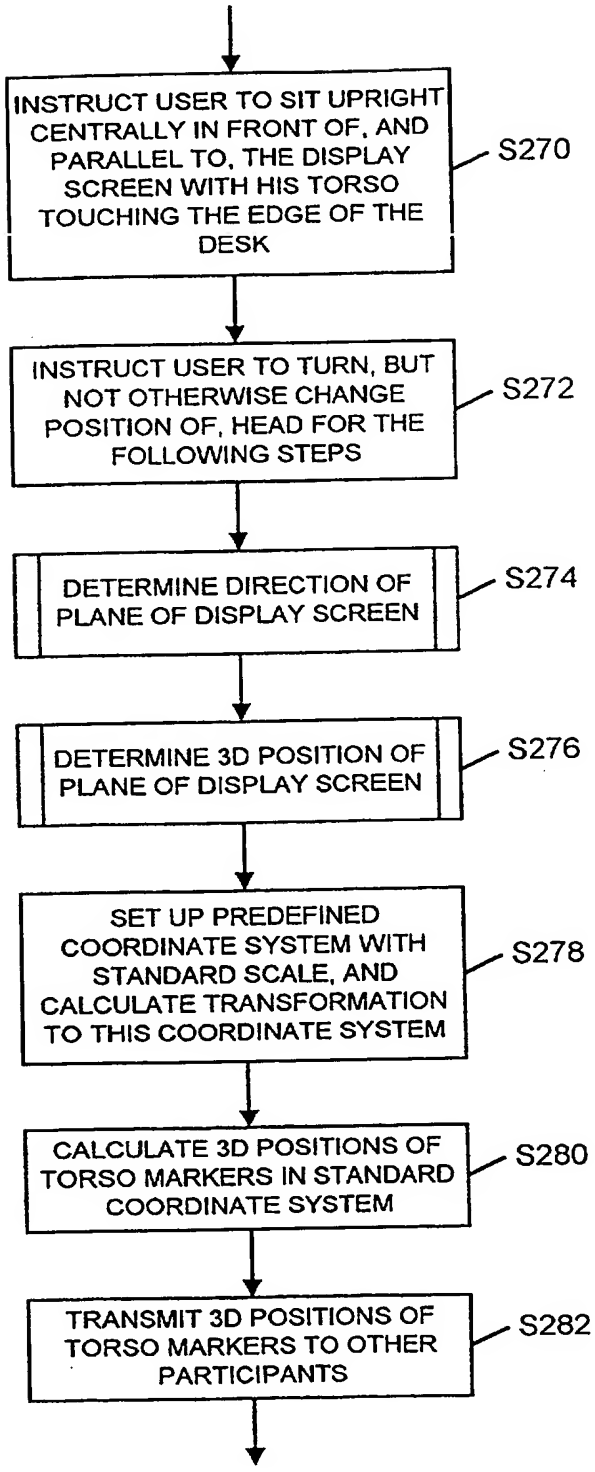

INSTRUCT USER TO SIT UPRIGHT CENTRALLY IN FRONT OF, AND PARALLEL TO, THE DISPLAY SCREEN WITH HIS TORSO TOUCHING THE EDGE OF THE DESK
S270
INSTRUCT USER TO TURN, BUT NOT OTHERWISE CHANGE POSITION OF, HEAD FOR THE FOLLOWING STEPS
S272
DETERMINE DIRECTION OF PLANE OF DISPLAY SCREEN
S274
DETERMINE 3D POSITION OF PLANE OF DISPLAY SCREEN
S276
SET UP PREDEFINED COORDINATE SYSTEM WITH STANDARD SCALE, AND CALCULATE TRANSFORMATION TO THIS COORDINATE SYSTEM
S278
CALCULATE 3D POSITIONS OF TORSO MARKERS IN STANDARD COORDINATE SYSTEM
S280
TRANSMIT 3D POSITIONS OF TORSO MARKERS TO OTHER PARTICIPANTS
S282

FIG. 17

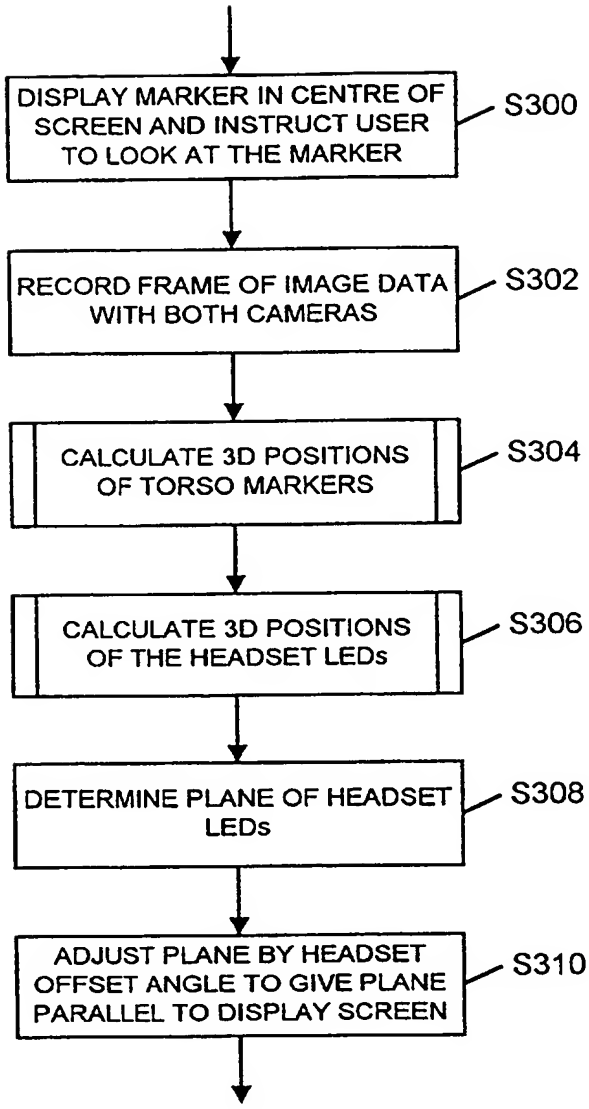

DISPLAY MARKER IN CENTRE OF SCREEN AND INSTRUCT USER TO LOOK AT THE MARKER
S300
RECORD FRAME OF IMAGE DATA WITH BOTH CAMERAS
S302
CALCULATE 3D POSITIONS OF TORSO MARKERS
S304
CALCULATE 3D POSITIONS OF THE HEADSET LEDs
S306
DETERMINE PLANE OF HEADSET LEDs
S308
ADJUST PLANE BY HEADSET OFFSET ANGLE TO GIVE PLANE PARALLEL TO DISPLAY SCREEN
S310

FIG. 18

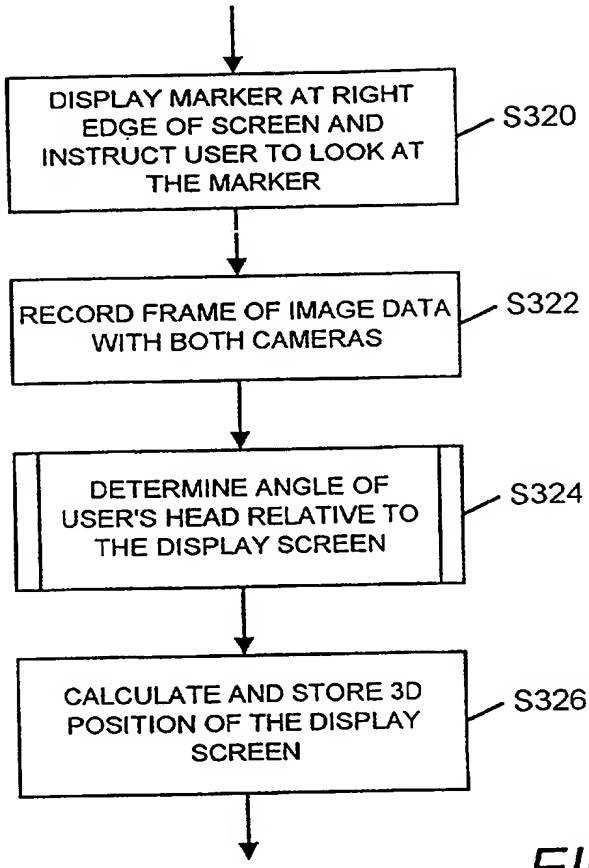

DISPLAY MARKER AT RIGHT EDGE OF SCREEN AND INSTRUCT USER TO LOOK AT THE MARKER
S320
RECORD FRAME OF IMAGE DATA WITH BOTH CAMERAS
S322
DETERMINE ANGLE OF USER'S HEAD RELATIVE TO THE DISPLAY SCREEN
S324
CALCULATE AND STORE 3D POSITION OF THE DISPLAY SCREEN
S326

FIG. 19

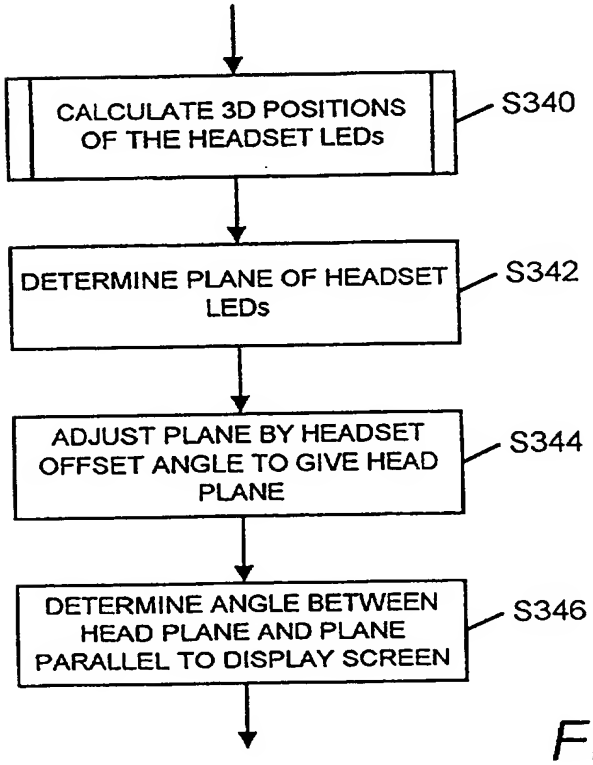
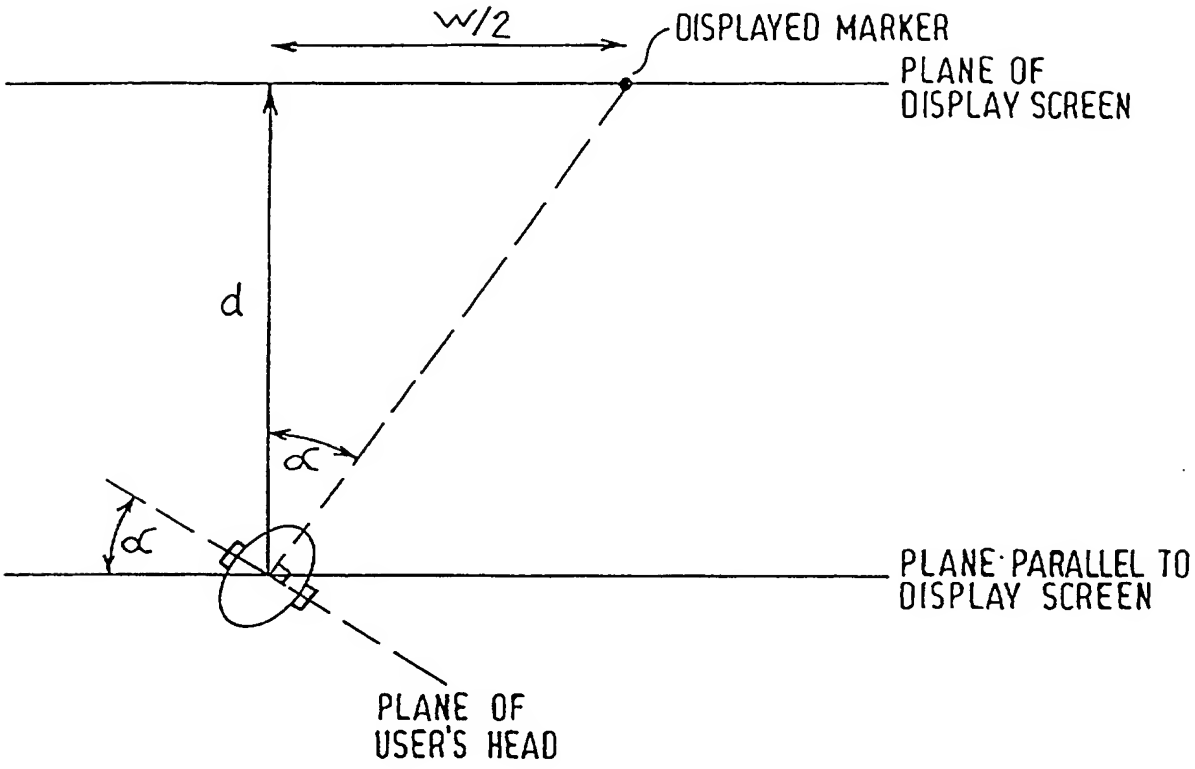

CALCULATE 3D POSITIONS OF THE HEADSET LEDs
S340
DETERMINE PLANE OF HEADSET LEDs
S342
ADJUST PLANE BY HEADSET OFFSET ANGLE TO GIVE HEAD PLANE
S344
DETERMINE ANGLE BETWEEN HEAD PLANE AND PLANE PARALLEL TO DISPLAY SCREEN
S346

FIG. 20

FIG. 21


W/2
DISPLAYED MARKER
PLANE OF
DISPLAY SCREEN
d
α
α
PLANE PARALLEL TO
DISPLAY SCREEN
PLANE OF
USER'S HEAD

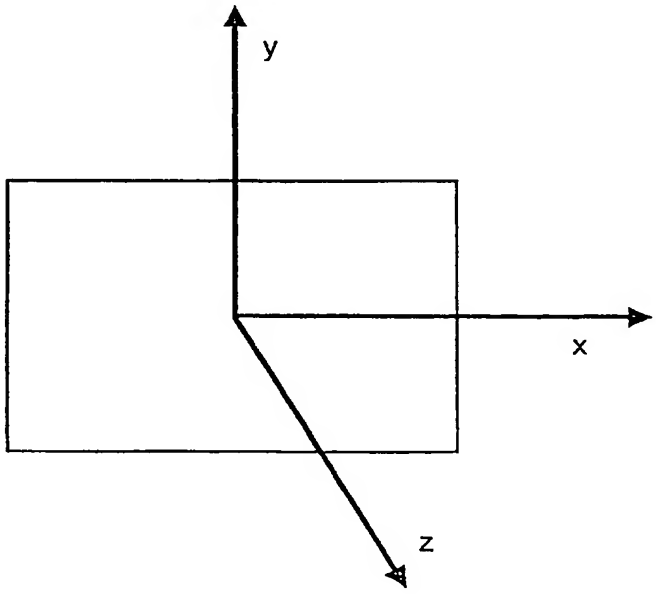

y
x
z

FIG. 22

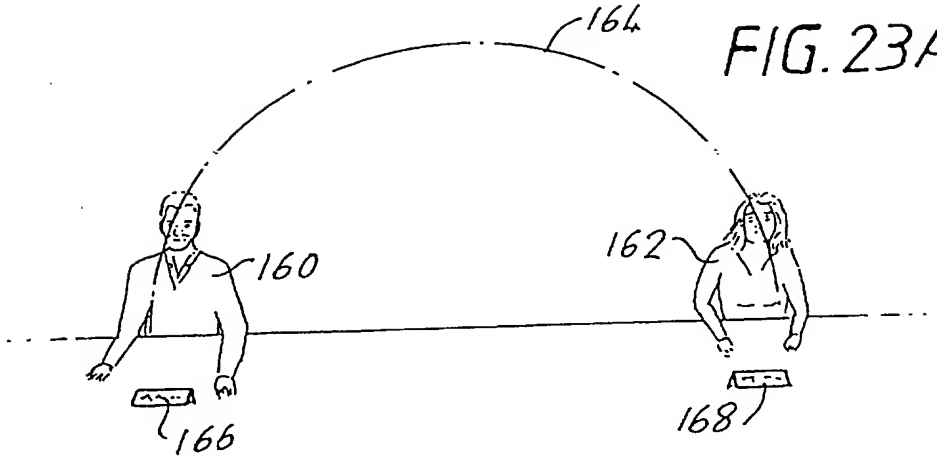


FIG. 23A

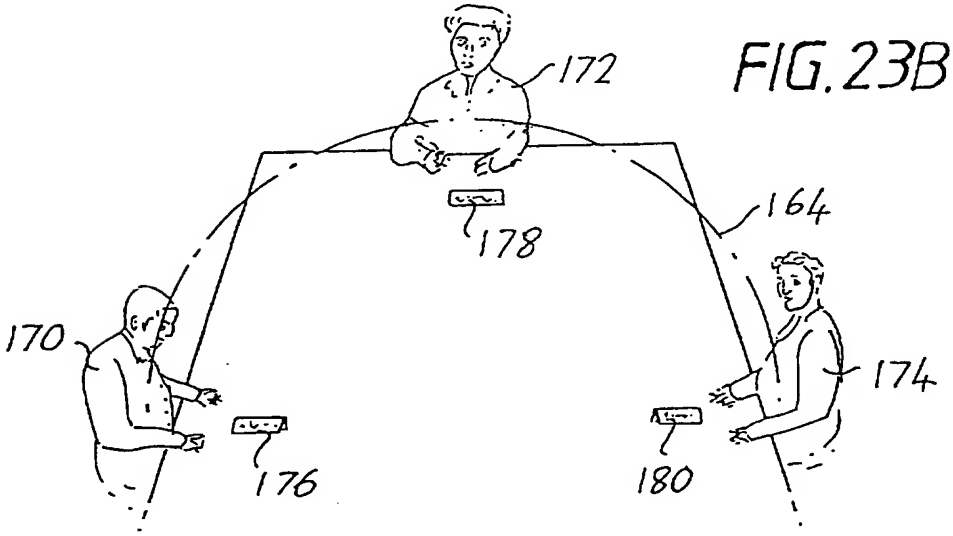


FIG. 23B

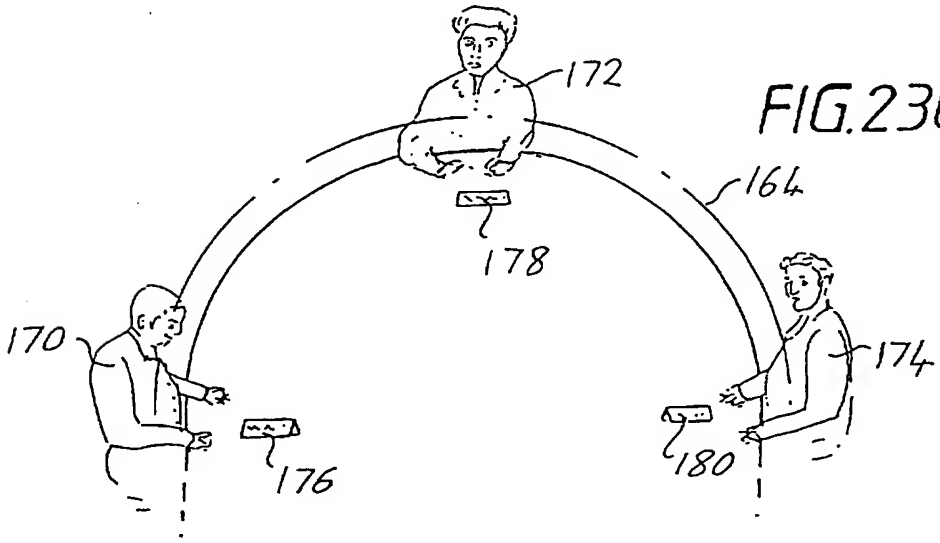


FIG. 23C

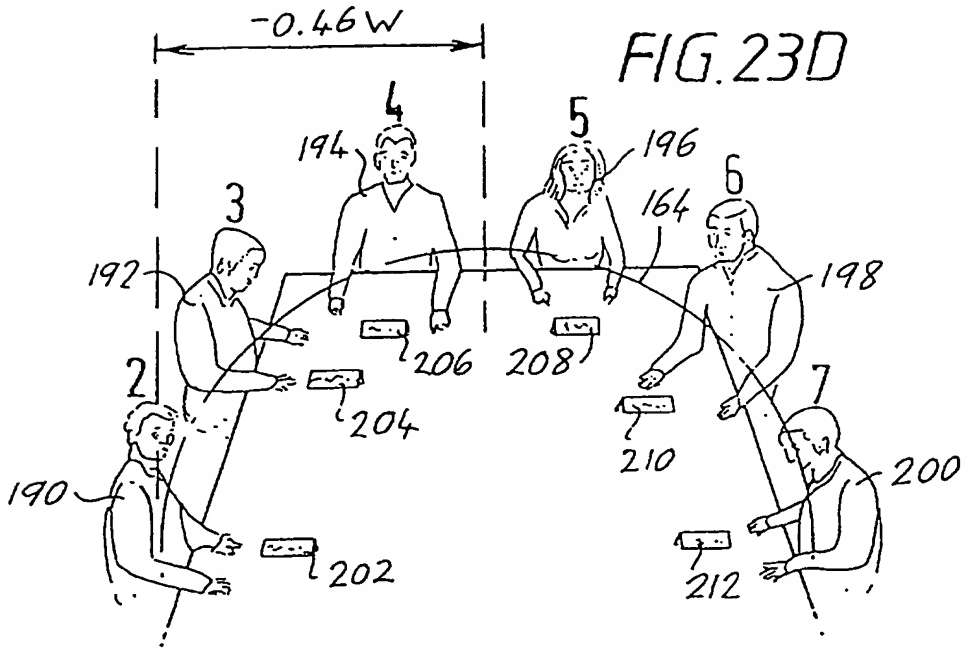


FIG. 23D

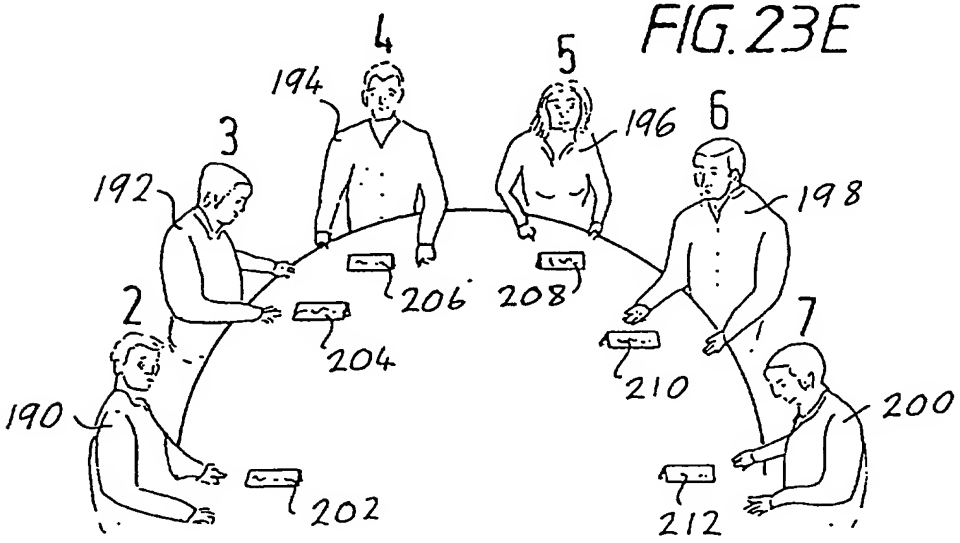


FIG. 23E

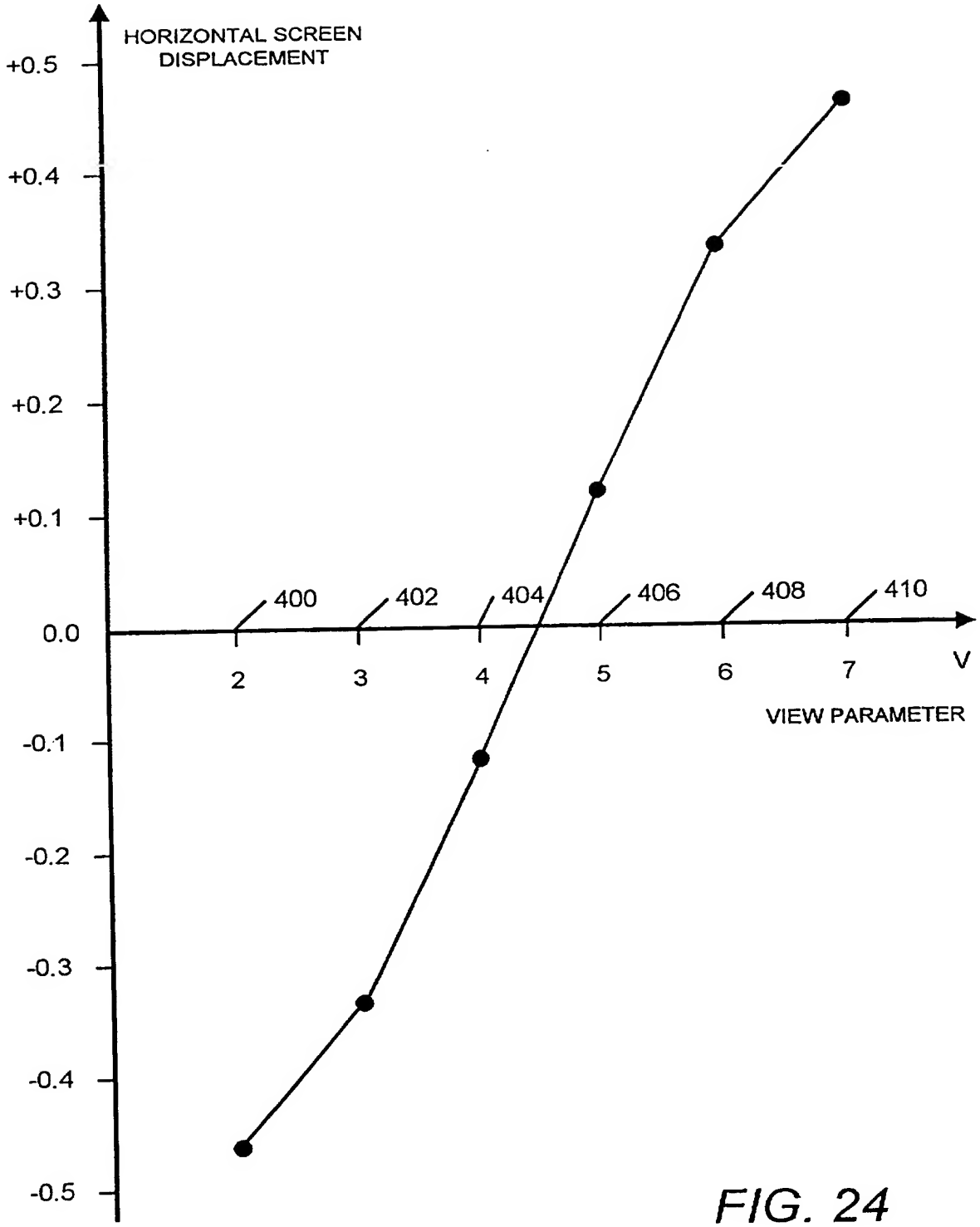

HORIZONTAL SCREEN DISPLACEMENT
+0.5
+0.4
+0.3
+0.2
+0.1
0.0
-0.1
-0.2
-0.3
-0.4
-0.5
400
402
404
406
408
410
2
3
4
5
6
7
V
VIEW PARAMETER

FIG. 24

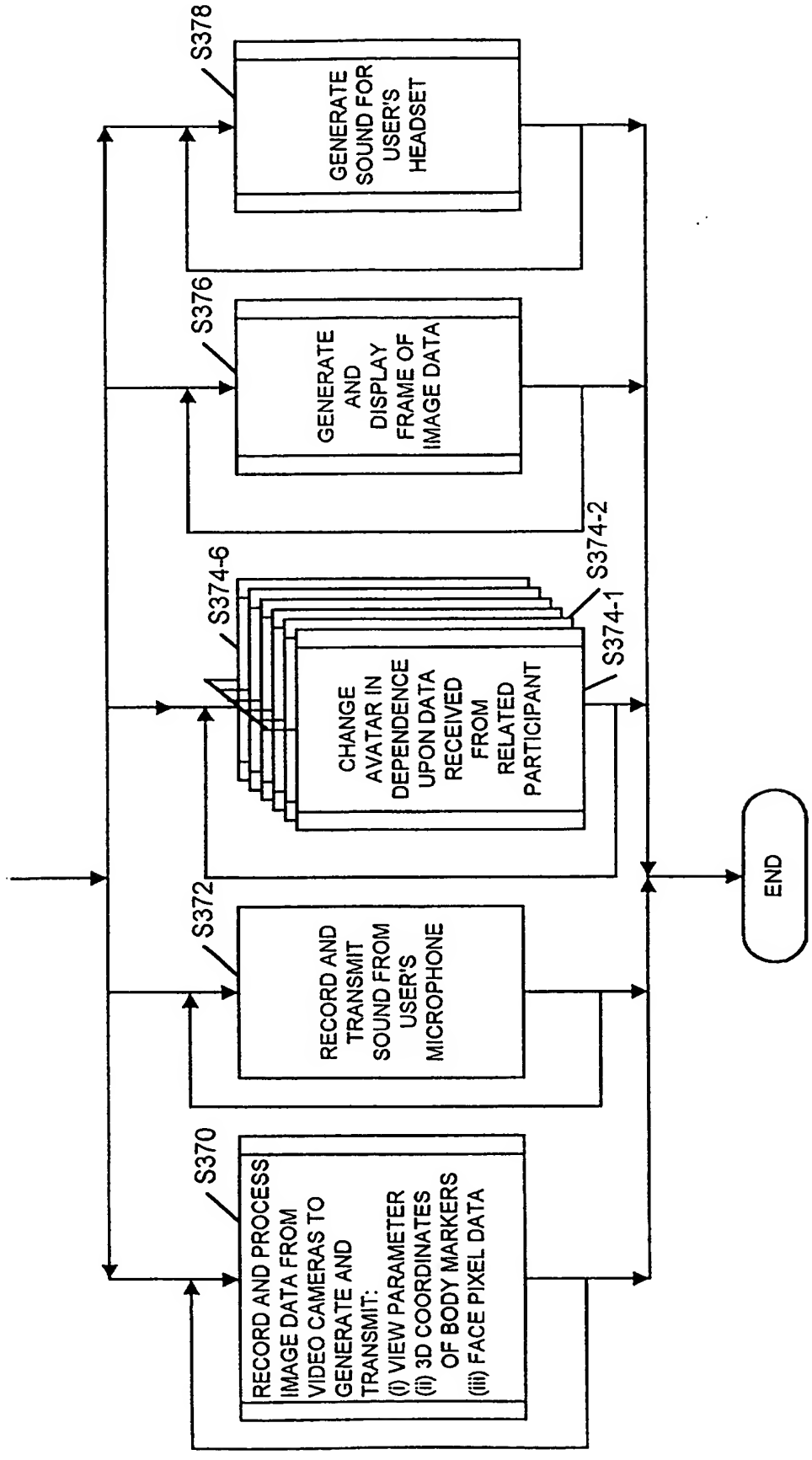

S370
RECORD AND PROCESS IMAGE DATA FROM VIDEO CAMERAS TO GENERATE AND TRANSMIT:
(i) VIEW PARAMETER
(ii) 3D COORDINATES OF BODY MARKERS
(iii) FACE PIXEL DATA
S372
RECORD AND TRANSMIT SOUND FROM USER'S MICROPHONE
S374-6
S374-2
S374-1
CHANGE AVATAR IN DEPENDENCE UPON DATA RECEIVED FROM RELATED PARTICIPANT
S376
GENERATE AND DISPLAY FRAME OF IMAGE DATA
S378
GENERATE SOUND FOR USER'S HEADSET
END

FIG. 25

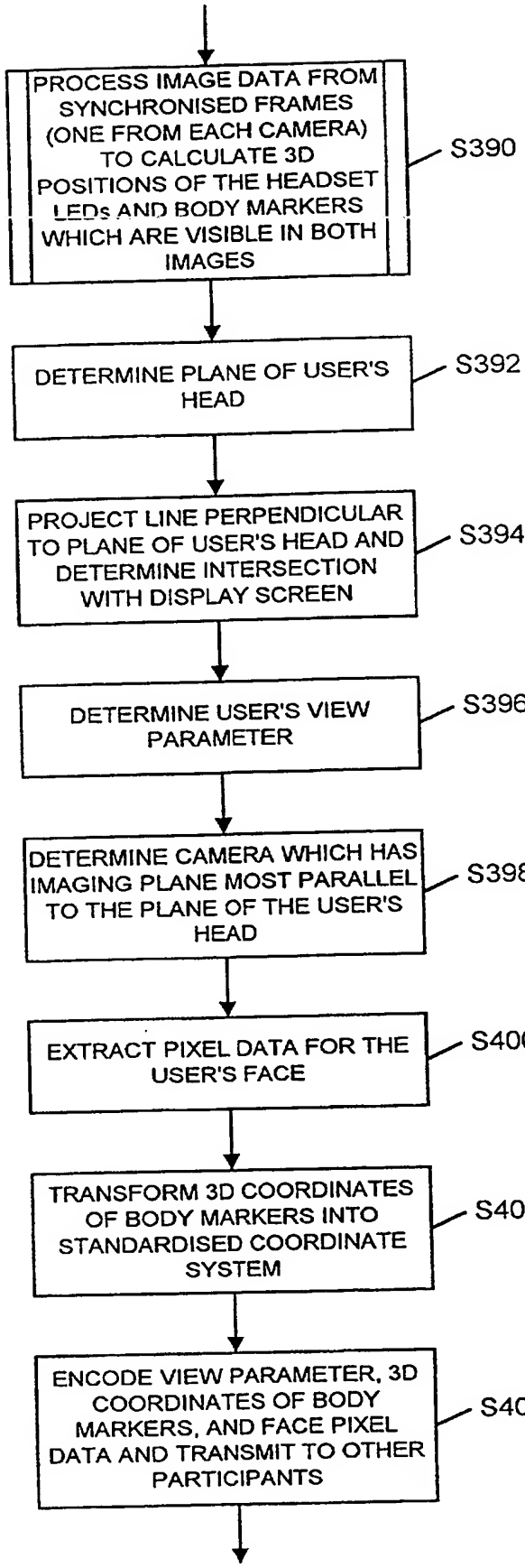

PROCESS IMAGE DATA FROM SYNCHRONISED FRAMES (ONE FROM EACH CAMERA) TO CALCULATE 3D POSITIONS OF THE HEADSET LEDs AND BODY MARKERS WHICH ARE VISIBLE IN BOTH IMAGES
S390
DETERMINE PLANE OF USER'S HEAD
S392
PROJECT LINE PERPENDICULAR TO PLANE OF USER'S HEAD AND DETERMINE INTERSECTION WITH DISPLAY SCREEN
S394
DETERMINE USER'S VIEW PARAMETER
S396
DETERMINE CAMERA WHICH HAS IMAGING PLANE MOST PARALLEL TO THE PLANE OF THE USER'S HEAD
S398
EXTRACT PIXEL DATA FOR THE USER'S FACE
S400
TRANSFORM 3D COORDINATES OF BODY MARKERS INTO STANDARDISED COORDINATE SYSTEM
S401
ENCODE VIEW PARAMETER, 3D COORDINATES OF BODY MARKERS, AND FACE PIXEL DATA AND TRANSMIT TO OTHER PARTICIPANTS
S402

FIG. 26

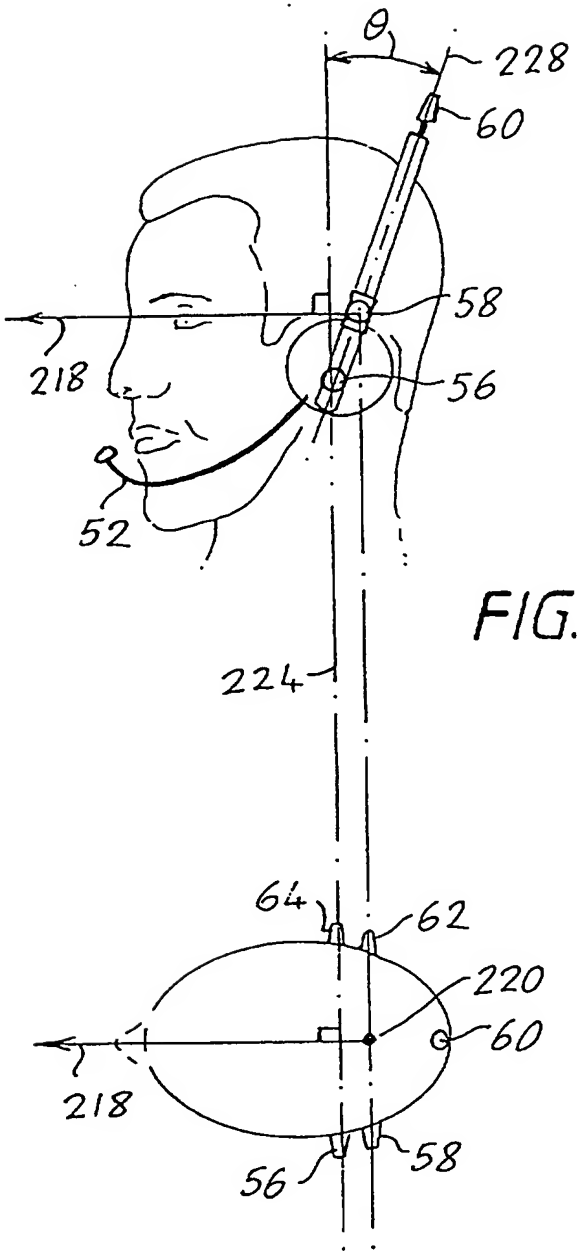


FIG. 27A

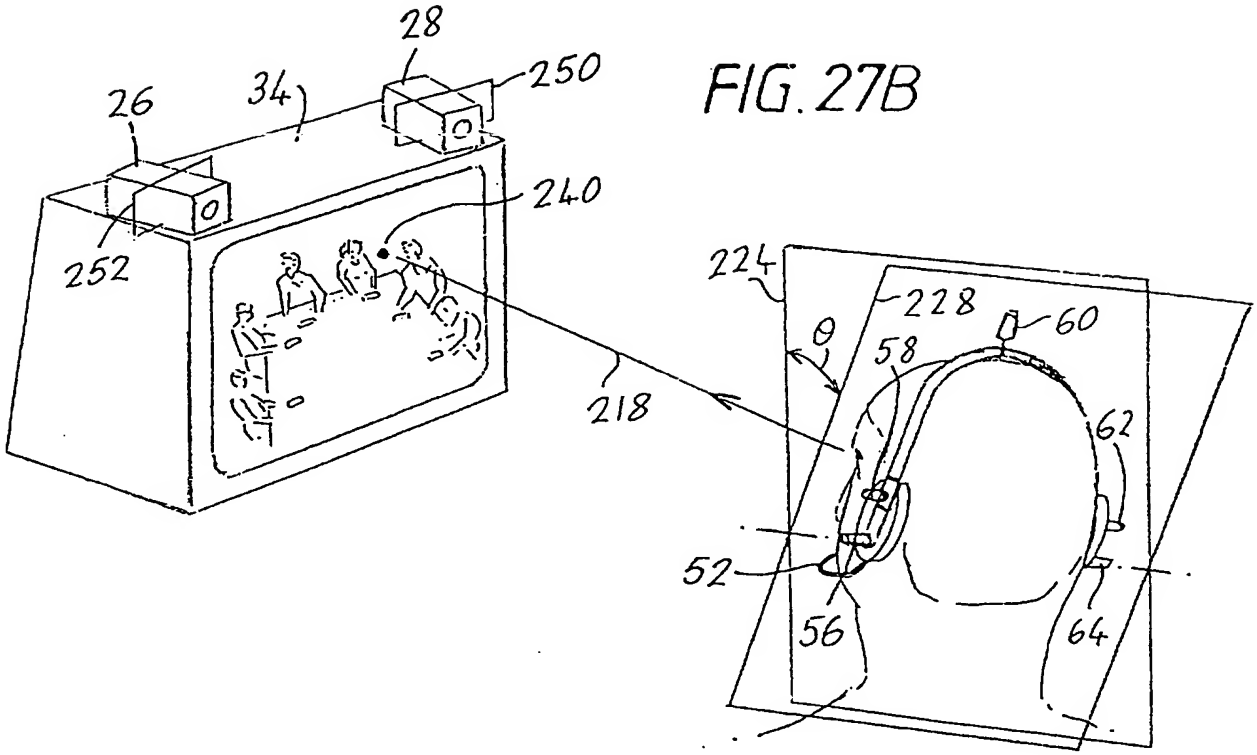
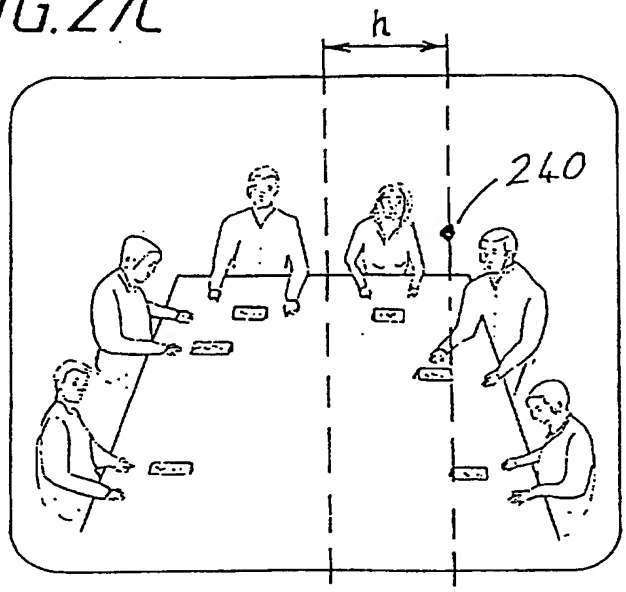


FIG. 27B

FIG. 27C



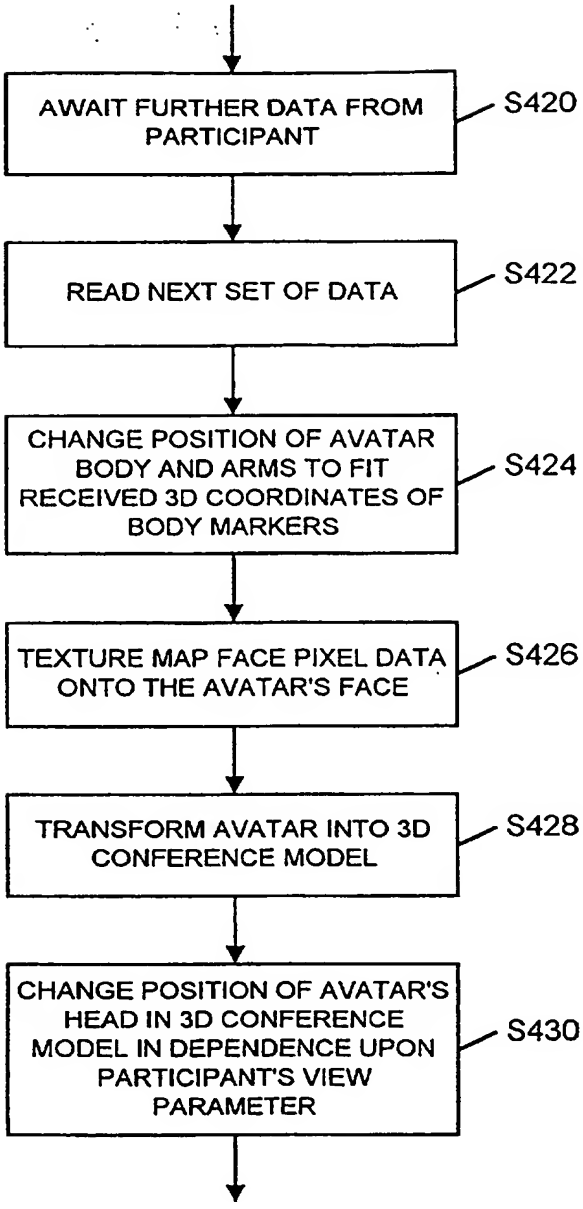

AWAIT FURTHER DATA FROM PARTICIPANT
S420
READ NEXT SET OF DATA
S422
CHANGE POSITION OF AVATAR BODY AND ARMS TO FIT RECEIVED 3D COORDINATES OF BODY MARKERS
S424
TEXTURE MAP FACE PIXEL DATA ONTO THE AVATAR'S FACE
S426
TRANSFORM AVATAR INTO 3D CONFERENCE MODEL
S428
CHANGE POSITION OF AVATAR'S HEAD IN 3D CONFERENCE MODEL IN DEPENDENCE UPON PARTICIPANT'S VIEW PARAMETER
S430

FIG. 28

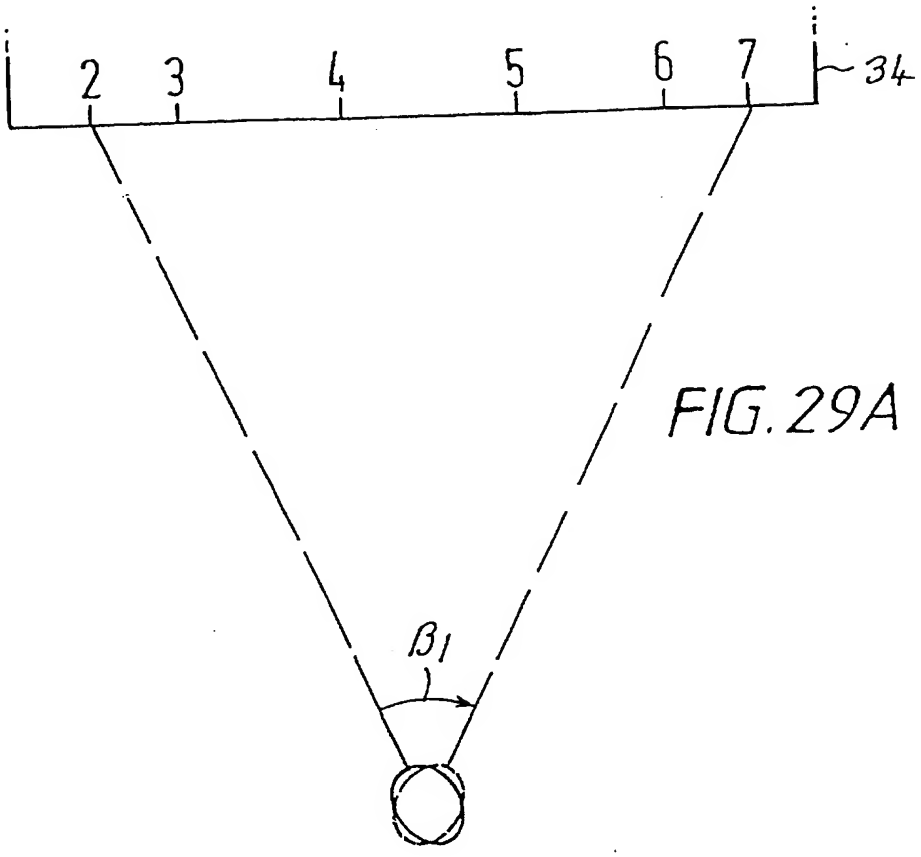


FIG. 29A

FIG. 29B

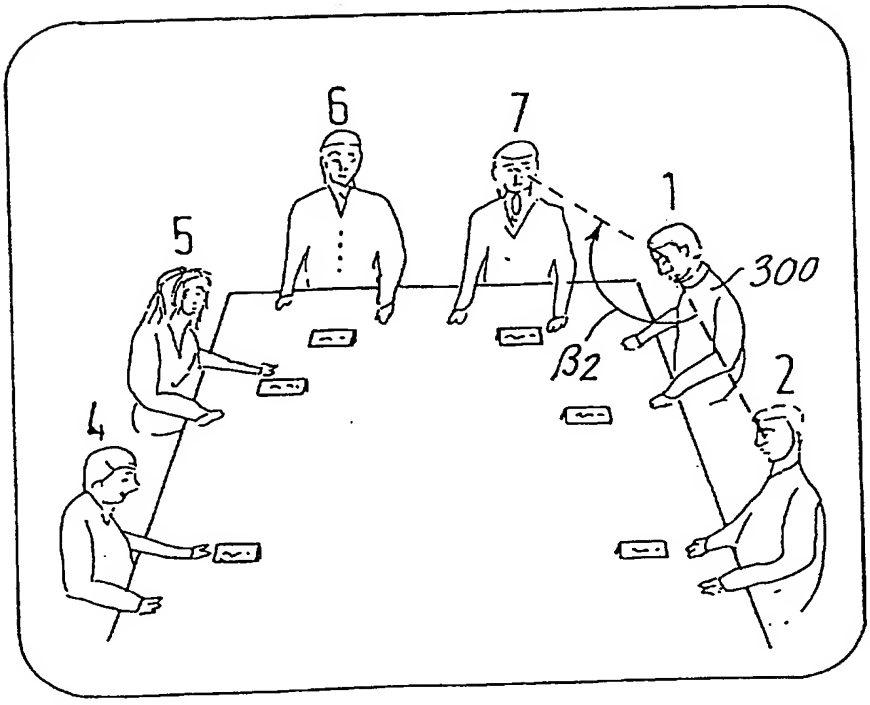
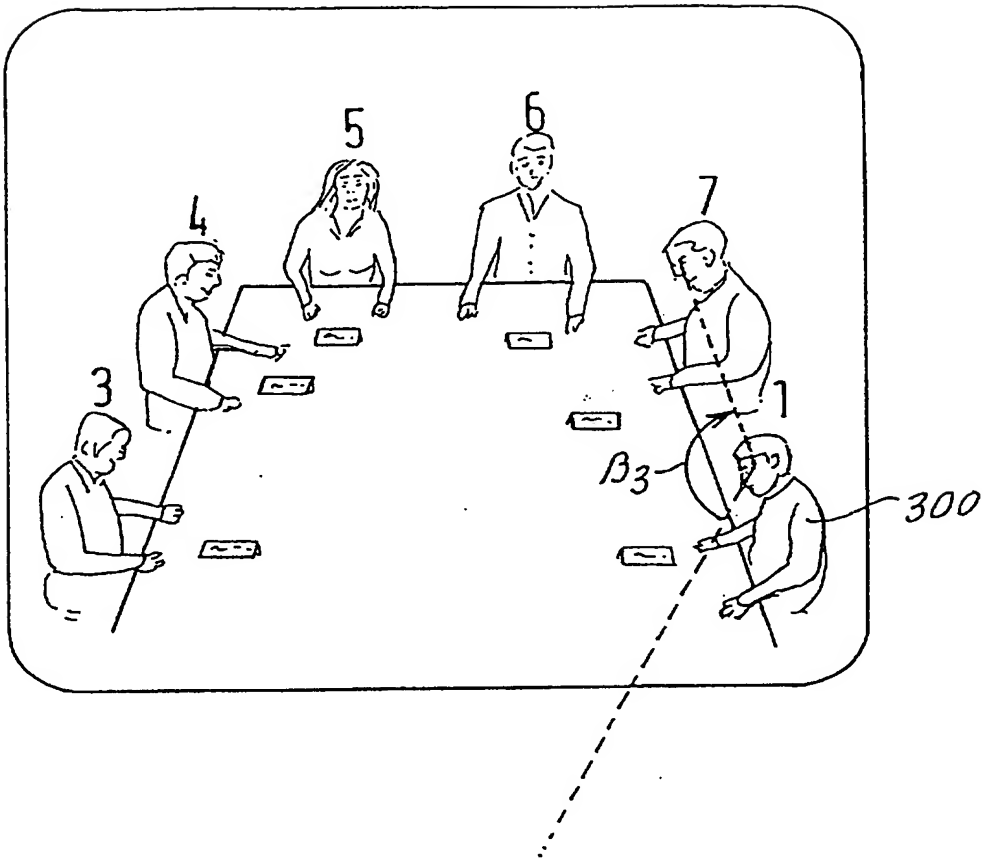


FIG. 29C



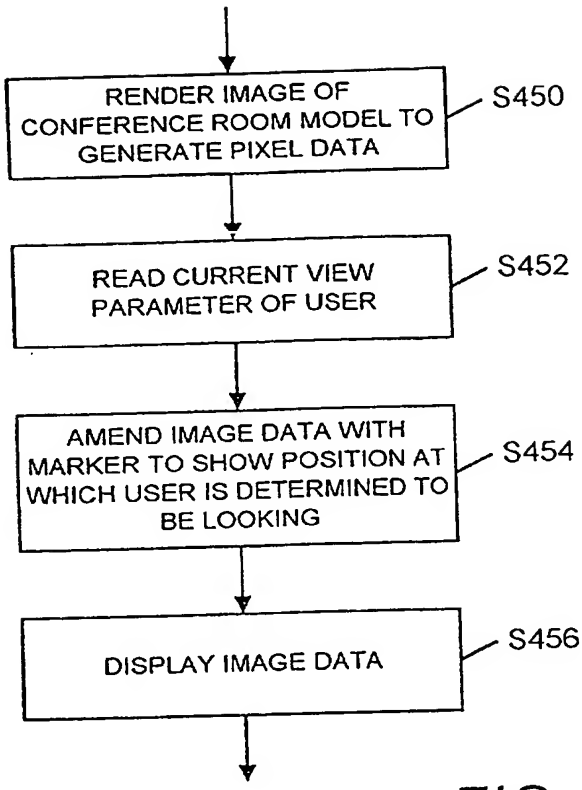

RENDER IMAGE OF CONFERENCE ROOM MODEL TO GENERATE PIXEL DATA
S450
READ CURRENT VIEW PARAMETER OF USER
S452
AMEND IMAGE DATA WITH MARKER TO SHOW POSITION AT WHICH USER IS DETERMINED TO BE LOOKING
S454
DISPLAY IMAGE DATA
S456

FIG. 30

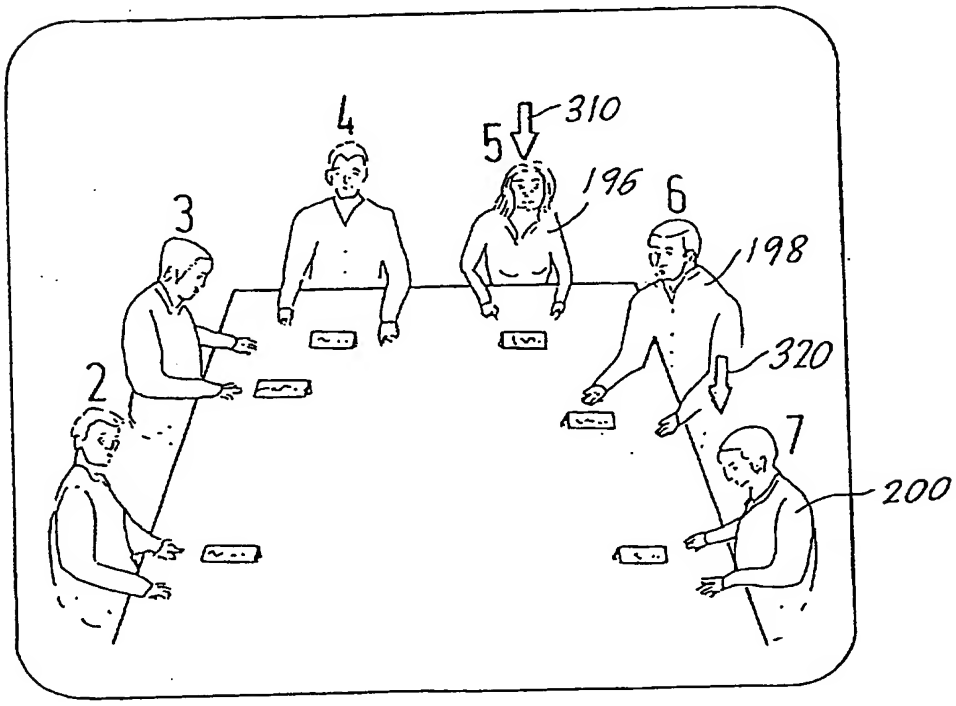

4
5
310
196
6
3
198
320
2
7
200

FIG. 31

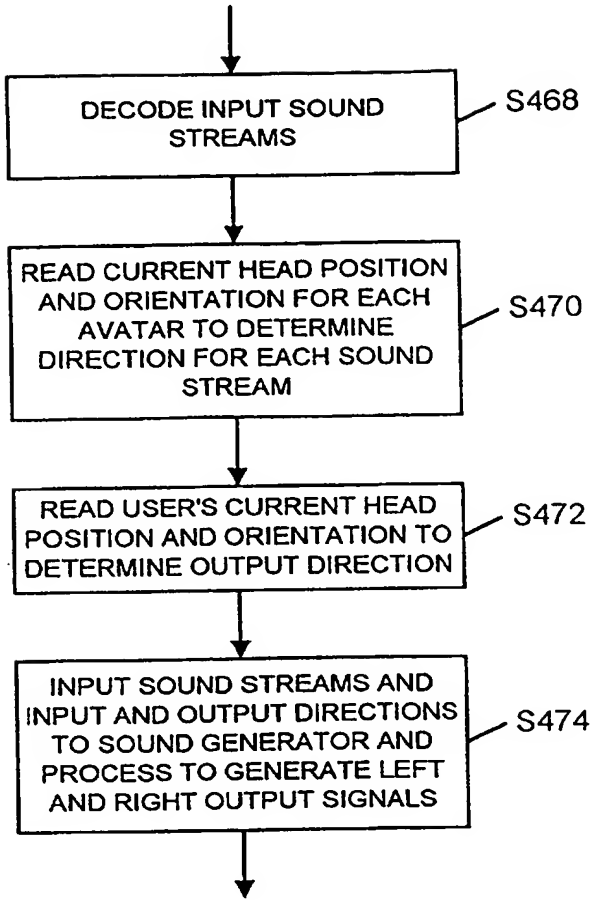

DECODE INPUT SOUND STREAMS
S468
READ CURRENT HEAD POSITION AND ORIENTATION FOR EACH AVATAR TO DETERMINE DIRECTION FOR EACH SOUND STREAM
S470
READ USER'S CURRENT HEAD POSITION AND ORIENTATION TO DETERMINE OUTPUT DIRECTION
S472
INPUT SOUND STREAMS AND INPUT AND OUTPUT DIRECTIONS TO SOUND GENERATOR AND PROCESS TO GENERATE LEFT AND RIGHT OUTPUT SIGNALS
S474

FIG. 32

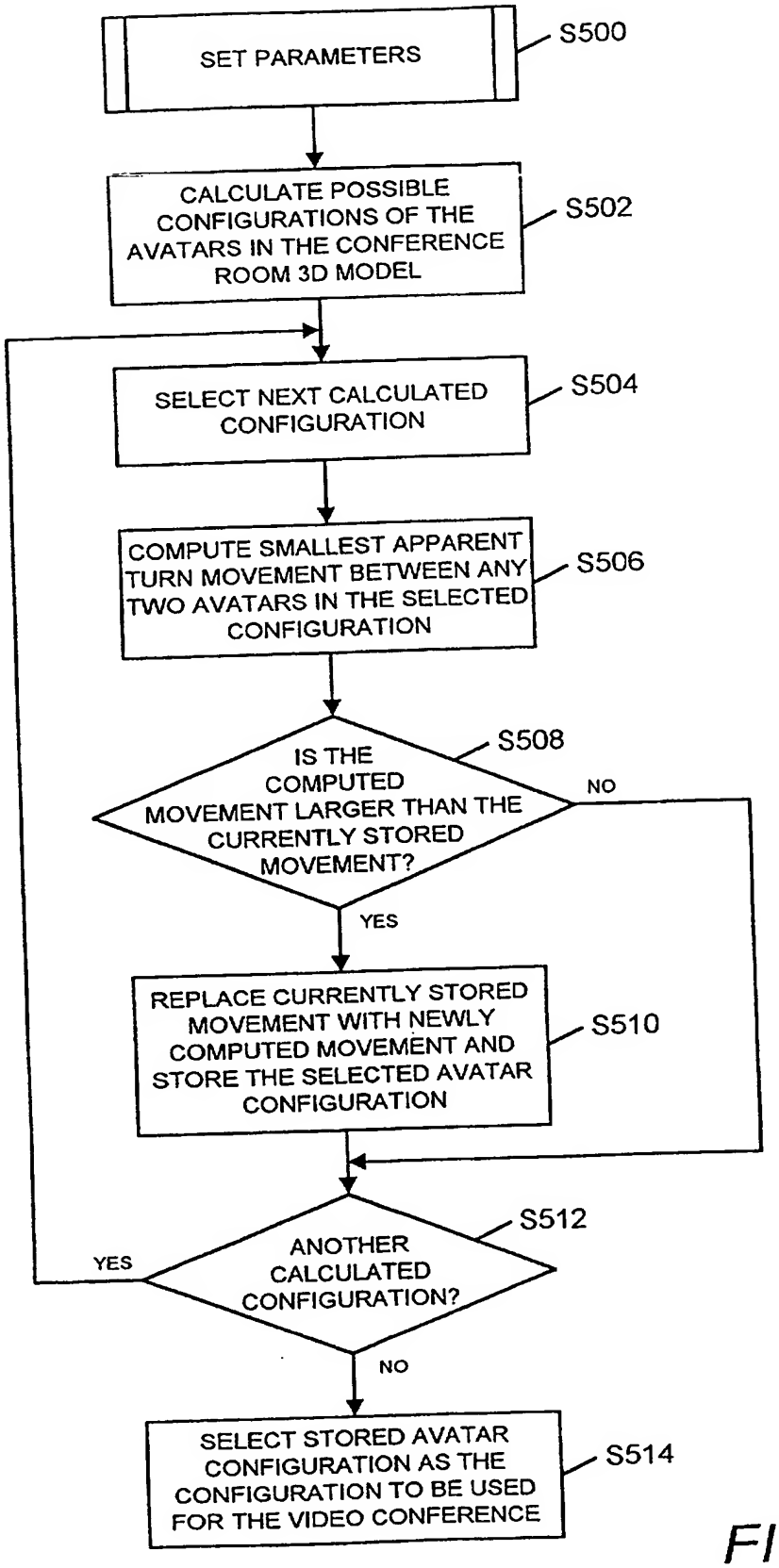

SET PARAMETERS
S500
CALCULATE POSSIBLE CONFIGURATIONS OF THE AVATARS IN THE CONFERENCE ROOM 3D MODEL
S502
SELECT NEXT CALCULATED CONFIGURATION
S504
COMPUTE SMALLEST APPARENT TURN MOVEMENT BETWEEN ANY TWO AVATARS IN THE SELECTED CONFIGURATION
S506
IS THE COMPUTED MOVEMENT LARGER THAN THE CURRENTLY STORED MOVEMENT?
S508
NO
YES
REPLACE CURRENTLY STORED MOVEMENT WITH NEWLY COMPUTED MOVEMENT AND STORE THE SELECTED AVATAR CONFIGURATION
S510
ANOTHER CALCULATED CONFIGURATION?
S512
YES
NO
SELECT STORED AVATAR CONFIGURATION AS THE CONFIGURATION TO BE USED FOR THE VIDEO CONFERENCE
S514

FIG. 33

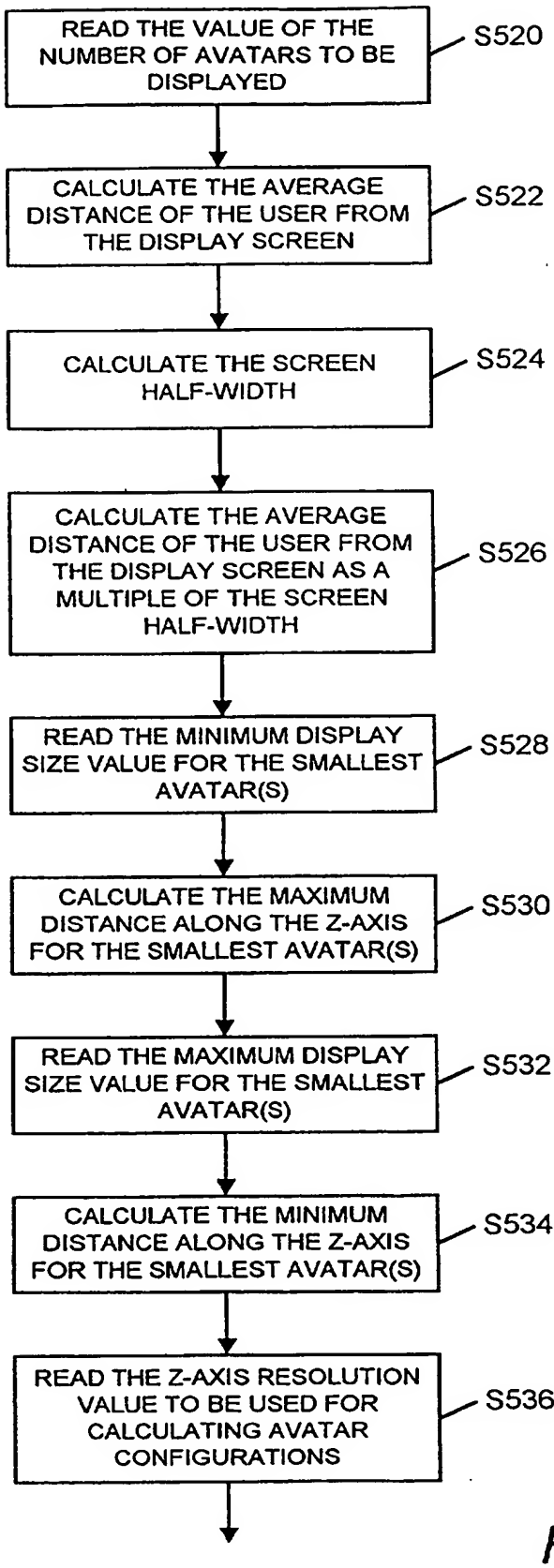

READ THE VALUE OF THE NUMBER OF AVATARS TO BE DISPLAYED
S520
CALCULATE THE AVERAGE DISTANCE OF THE USER FROM THE DISPLAY SCREEN
S522
CALCULATE THE SCREEN HALF-WIDTH
S524
CALCULATE THE AVERAGE DISTANCE OF THE USER FROM THE DISPLAY SCREEN AS A MULTIPLE OF THE SCREEN HALF-WIDTH
S526
READ THE MINIMUM DISPLAY SIZE VALUE FOR THE SMALLEST AVATAR(S)
S528
CALCULATE THE MAXIMUM DISTANCE ALONG THE Z-AXIS FOR THE SMALLEST AVATAR(S)
S530
READ THE MAXIMUM DISPLAY SIZE VALUE FOR THE SMALLEST AVATAR(S)
S532
CALCULATE THE MINIMUM DISTANCE ALONG THE Z-AXIS FOR THE SMALLEST AVATAR(S)
S534
READ THE Z-AXIS RESOLUTION VALUE TO BE USED FOR CALCULATING AVATAR CONFIGURATIONS
S536

FIG. 34

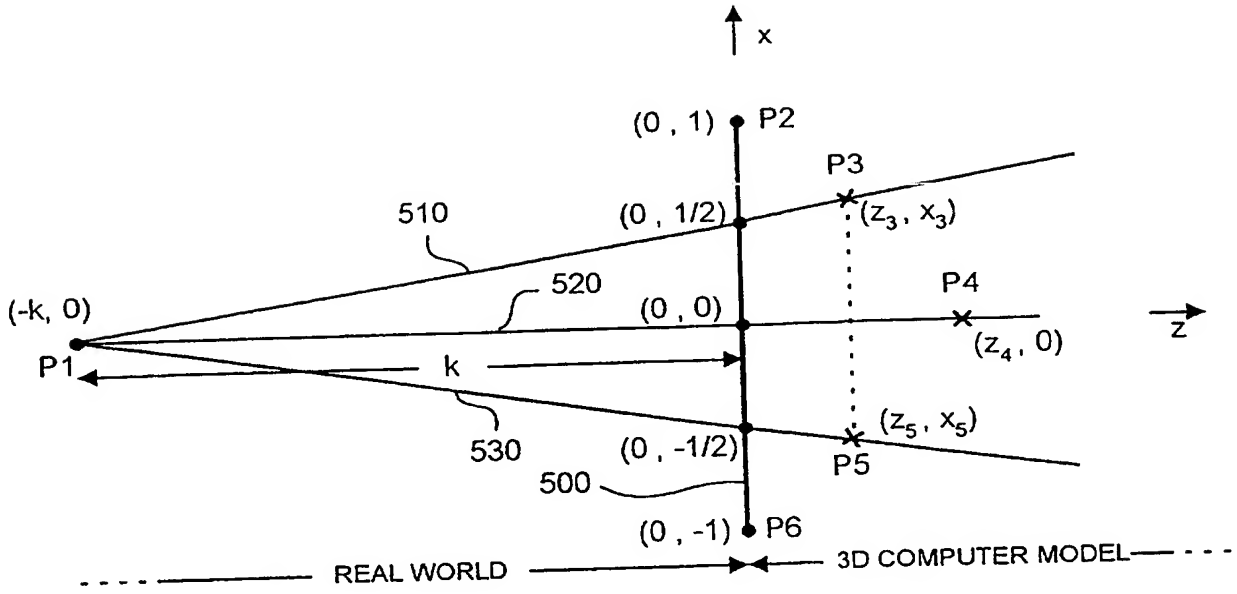


FIG. 35A

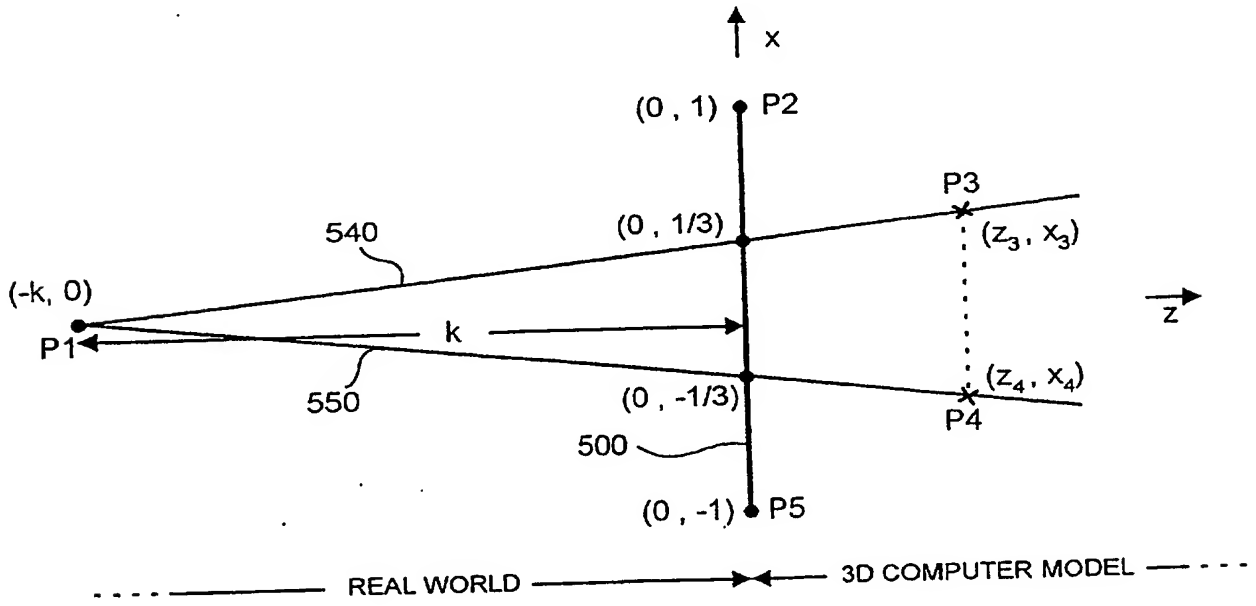


FIG. 35B

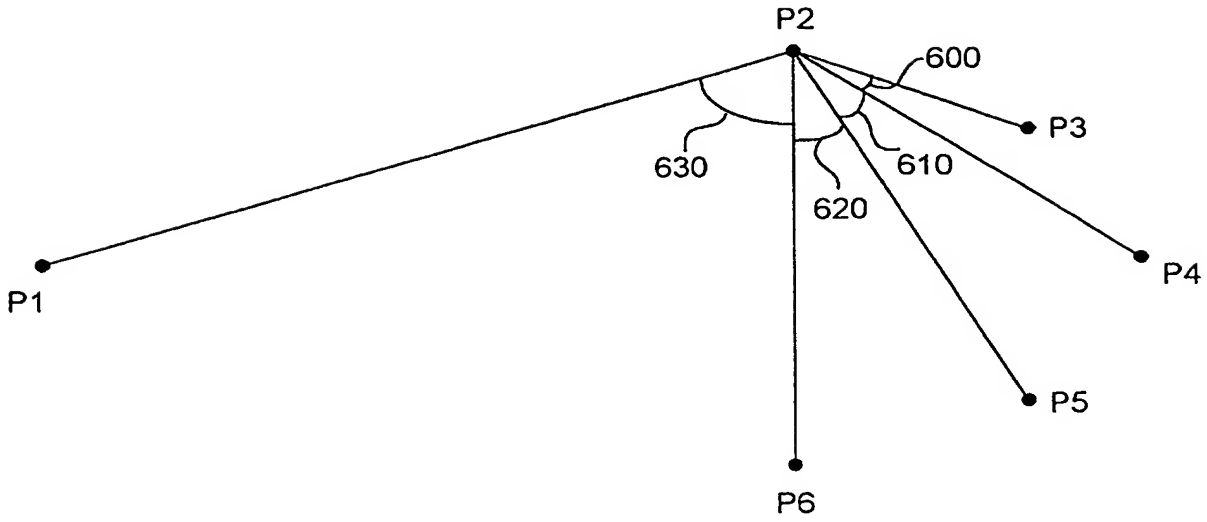


FIG. 36A

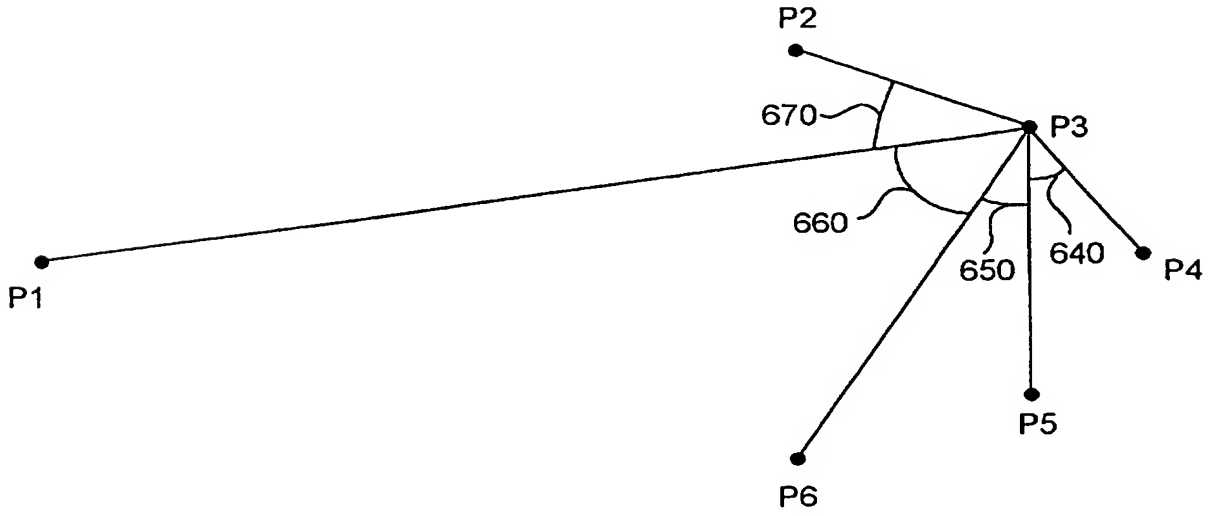


FIG. 36B

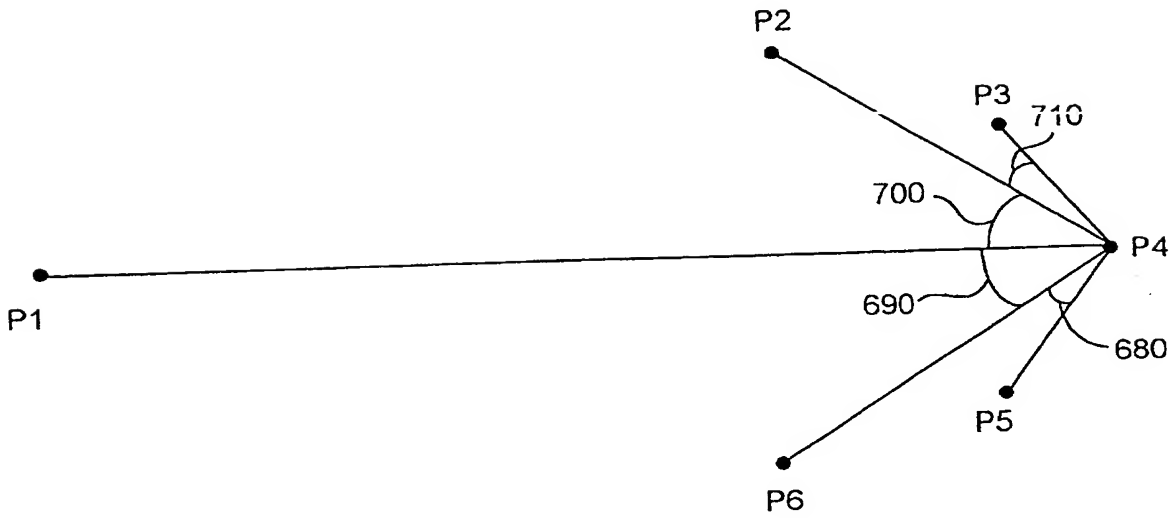


FIG. 36C

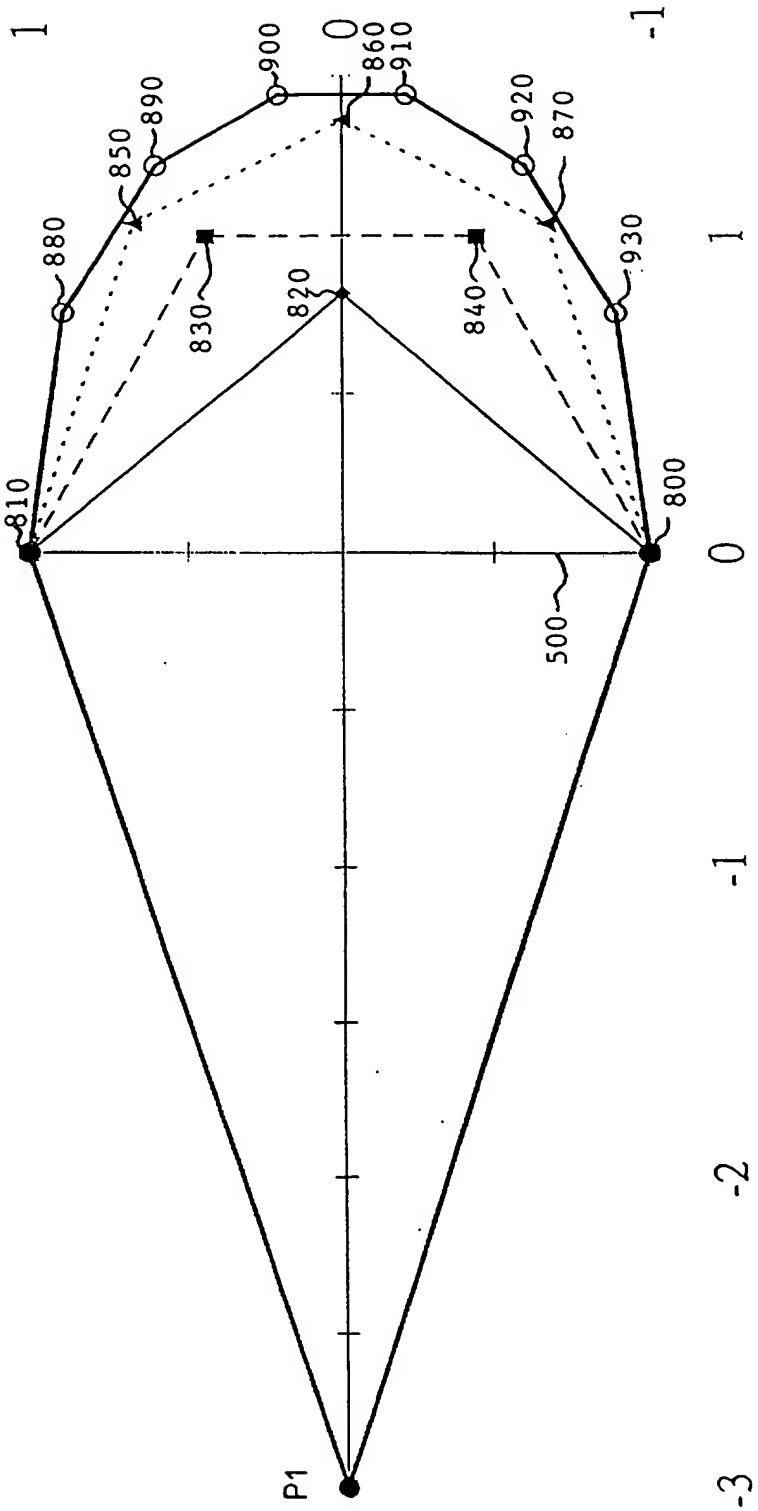

1
0
-1
900
860
910
890
850
920
870
880
930
830
820
840
810
800
500
P1
1
0
-1
-2
-3

FIG. 37

Number of avatars to be displayed	Distance of viewer from display, measured as a proportion of the display half-width			
	2: (-2, 0)	3: (-3, 0)	4: (-4, 0)	5: (-5, 0)
3	0 1 0.78 0 0 -1	0 1 0.82 0 0 -1	0 1 0.86 0 0 -1	0 1 0.88 0 0 -1
4	0 1 1.04 0.50667 1.04 -0.50667 0 -1	0 1 1 0.44444 1 -0.44444 0 -1	0 1 1.12 0.42667 1.12 -0.42667 0 -1	0 1 1.14 0.40933 1.14 -0.40933 0 -1
5	0 1 1.08 0.77 1.46 0 1.08 -0.77 0 -1	0 1 1.04 0.67333 1.36 0 1.04 -0.67333 0 -1	0 1 1.02 0.6275 1.28 0 1.02 -0.6275 0 -1	0 1 0.9 0.59 1.12 0 0.9 -0.59 0 -1
6	0 1 1.02 0.906 1.56 0.356 1.56 -0.356 1.02 -0.906 0 -1	0 1 0.9 0.78 1.32 0.288 1.32 -0.288 0.9 -0.78 0 -1	0 1 0.94 0.741 1.3 0.265 1.3 -0.265 0.94 -0.741 0 -1	0 1 0.94 0.7128 1.26 0.2504 1.26 -0.2504 0.94 -0.7128 0 -1
7	0 1 0.82 0.94 1.34 0.55667 1.52 0 1.34 -0.55667 0.82 -0.94 0 -1	0 1 0.94 0.87556 1.44 0.49333 1.6 0 1.44 -0.49333 0.94 -0.87556 0 -1	0 1 0.8 0.8 1.18 0.43167 1.3 0 1.18 -0.43167 0.8 -0.8 0 -1	0 1 0.76 0.768 1.1 0.40667 1.2 0 1.1 -0.40667 0.76 -0.768 0 -1
8	0 1 0.7 0.96429 1.2 0.68571 1.46 0.24714 1.46 -0.24714 1.2 -0.68571 0.7 -0.96429 0 -1	0 1 0.76 0.89524 1.22 0.60286 1.44 0.21143 1.44 -0.21143 1.22 -0.60286 0.76 -0.89524 0 -1	0 1 0.76 0.85 1.16 0.55286 1.34 0.19071 1.34 -0.19071 1.16 -0.55286 0.76 -0.85 0 -1	0 1 0.8 0.82857 1.16 0.528 1.32 0.18057 1.32 -0.18057 1.16 -0.528 0.8 -0.82857 0 -1

1000

FIG. 38

COMPUTER CONFERENCING APPARATUS

The present invention relates to the field of remote conferencing, and more particularly, to computer conferencing carried out by animating three-dimensional computer models (avatars) in dependence upon real-life movements of the conference participants.

A number of systems for carrying out remote conferences, such as video conferences, are known.

In a typical, conventional system, a camera records images of one or more conference participants and the image data is sent to the other participants, where it is displayed. In conferences involving participants at three or more sites, data is displayed to a given one of the participants by displaying the video images of the participants from the other sites side by side on a display screen. This type of system suffers from a number of problems, however. For example, the direction of a participant's gaze (that is, where the participant is looking) and his body gestures cannot be accurately communicated to the other participants. More particularly, if a participant turns his head, or points, to the participant displayed on the right of his screen, then the other participants see the user move his head, or point, to the right but do not know how this movement relates to the other participants in the conference. Accordingly, it is not possible to reproduce eye contact and body gestures, which have been shown to be necessary cues for effective communication.

A common solution to the problem of communicating gaze and gestures is to

perform the video conference in a virtual space which is shared by all of the participants. Each participant is represented by an avatar (that is, a three-dimensional computer model) in the virtual space, and the avatars are then animated using motion parameters measured from the motion of the real participants. In this way, a participant can move around the conference room by transmitting the necessary motion parameters to his avatar. Images of the avatars in the virtual space are displayed to each participant so that a simulated video conference is seen. Methods which have been suggested for image display in such a system include displaying the images on a large, life-size display so that the participants are positioned where they would have been if the meeting was real, for example as described in "Virtual space teleconferencing: real-time reproduction of 3D human images" by J. Ohya, Y. Kitamura, F. Kishino and N. Terashima in Journal of Visual Communication and Image Representation, 6(1), pages 1-25, March 1995. In this way, as the participant to whom the images are displayed moves his head, different parts of the screen become visible in the same way that different parts of the meeting room would become visible in a real-world conference. This method, however, suffers from the problem that the display is extremely large and expensive.

Another method which has been suggested for displaying images of a virtual conference is to display them on a conventional small screen display device, and to change the view displayed on the device to the participant as the direction of the eyes of the participant's avatar change in the virtual space, for example as described in US 5736982. It has also been suggested to display images in this way on a head-mounted display. These systems suffer from the

problem, however, that the user can find the displayed images confusing and unnatural. In addition, head-mounted displays are expensive and somewhat cumbersome.

A further approach to communicating gaze and gesture information is disclosed in "Interfaces for Multiparty Videoconferences" by Buxton et al in Video-Mediated Communication (Editors Finn, Sellen & Wilbur), Lawrence Erlbaum Associates, 1997, ISBN 0-8058-2288-7, pages 385-400. In this system, a virtual conference approach is not adopted, and instead video images of each participant are recorded and sent to the other participants. The video images for each participant are then displayed on separate display modules which are arranged around the viewer's desk in exactly the same positions that the participants would occupy in a real video conference. This system suffers from the problems that the number of display modules, and hence the cost, increases as the number of participants in the meeting increases, and also the process for arranging the modules in the correct positions is difficult and time consuming.

A further approach is disclosed in "Look Who's Talking: The GAZE Groupware System" by Vertegaal et al, in Summary of ACM CHI'98 Conference on Human Factors in Computing Systems, Los Angeles, April 1998, pages 293-294. In this system, a shared virtual meeting room is again proposed, but instead of avatars, a two-dimensional model of a display screen is positioned where each participant would sit. Images of the virtual room are then rendered from a unique, constant viewpoint for each participant. An eye tracking system is used to measure each participant's eye movements in real

life and a camera records snap shots of the user. The 2D display screen in the virtual meeting room for a participant is then moved according to the participant's eye movements by rotating it about one or two axes, and the snap shot image data is presented on the 2D display screen. This system, too, suffers from a number of problems. For example, the images displayed to each participant are unrealistic, and it becomes difficult to arrange the 2D display screens in the virtual conference room for any more than four conference participants.

The present invention has been made with the above problems in mind.

The present invention provides a computer conferencing system and method, and apparatus for use therein, in which gaze information is communicated by providing avatars of the participants in a different three-dimensional model at each participant apparatus, and by changing the view direction of each avatar using information defining where the corresponding participant is looking in real-life.

In this way, because the three-dimensional model is different at each participant apparatus, a participant's avatar undergoes different movements at each apparatus, and gaze information can be accurately conveyed.

The present invention also provides a system for conducting a virtual meeting comprising a plurality of apparatus for use by participants which are arranged to generate and exchange data such that rotations of a participant's head in real-life cause rotations of the head of a corresponding avatar which differ at

different apparatus.

Embodiments of the invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figure 1 schematically shows a plurality of user stations interconnected to carry out a video conference in an embodiment of the invention;

Figure 2A shows a user station and a user, Figure 2B shows the headset and body markers worn by the user, and Figure 2C shows the components of the headset worn by the user;

Figure 3 is a block diagram showing an example of notional functional components within the computer processing apparatus at each user station;

Figure 4 shows the steps performed to carry out a video conference;

Figure 5 shows the processing operations performed at step S4 in Figure 4;

Figure 6 shows an example seating plan defined at step S24 in Figure 5;

Figure 7 shows the processing operations performed at step S6 in Figure 4;

Figure 8 shows the processing operations performed at step S62 in Figure 7;

Figure 9 shows the processing operations performed at step S100 in Figure 8;

Figure 10 shows the processing operations performed at step S130 in Figure 9;

Figure 11 shows the processing operations performed at step S146 and step S150 in Figure 10;

Figure 12 shows the processing operations performed at step S132 in Figure 9;

Figure 13 illustrates the offset angle θ between the plane of the user's head and the plane of his headset calculated at step S64 in Figure 7;

Figure 14 shows the processing operations performed at step S64 in Figure 7;

Figure 15 shows the processing operations performed at step S234 in Figure 14;

Figure 16 illustrates the line projection and mid-point calculation performed at step S252 and step S254 in Figure 15;

Figure 17 shows the processing operations performed at step S66 in Figure 7;

Figure 18 shows the processing operations performed at step S274 in Figure 17;

Figure 19 shows the processing operations performed at step S276 in Figure 17;

Figure 20 shows the processing operations performed at step S324 in Figure 19;

Figure 21 illustrates the angle calculation performed at step S346 in Figure 20;

Figure 22 illustrates the standard coordinate system set up at step S278 in Figure 17;

Figures 23A, 23B, 23C, 23D and 23E show examples of avatar positions at conference room tables;

Figure 24 shows a piece-wise linear function relating horizontal screen position to view parameter, which is stored at step S72 in Figure 7;

Figure 25 shows the processing operations performed at step S8 in Figure 4;

Figure 26 shows the processing operations performed at step S370 in Figure 25;

Figures 27A, 27B and 27C illustrate the calculation at step S394 in Figure 26 of the point at which the user is looking by projecting a line from the plane of the user's head and determining the intersection of the line with the display screen;

Figure 28 shows the processing operations performed in each of steps S374-1 to S374-6 in Figure 25;

Figures 29A, 29B and 29C illustrate how the position of an avatar's head is changed in dependence upon changes of the corresponding participant's head in real-life at step S430 in Figure 28;

Figure 30 shows the processing operations performed at step S376 in Figure 25;

Figure 31 illustrates examples of markers displayed in images at steps S454 and S456 in Figure 30;

Figure 32 shows the processing operations performed at step S378 in Figure 25;

Figure 33 shows the processing operations performed in a modification to calculate the positions of the avatars in a 3D computer model of the conference room such that, when an image of the conference room model is displayed on the display screen, the avatars are evenly spaced across a horizontal line on the display and the minimum movement which the head of an avatar appears to undergo to look from one avatar to another is maximised;

Figure 34 shows the processing operations performed at step S500 in Figure 33;

Figures 35A and 35B schematically illustrate the processing performed at step S502 in Figure 33;

Figures 36A, 36B and 36C schematically illustrate the processing performed at step S506 in Figure 33;

Figure 37 shows examples of the results of performing the processing at steps S500 to S512 in Figure 33; and

Figure 38 shows an example of a look-up-table and the data stored therein which may be stored at a user station to determine the positions of avatars in the 3D computer model of the conference room at the user station.

Referring to Figure 1, in this embodiment, a plurality of user stations 2, 4, 6, 8, 10, 12, 14 are connected via a communication path 20, such as the Internet, wide area network (WAN), etc.

As will be described below, each user station 2, 4, 6, 8, 10, 12, 14 comprises apparatus to facilitate a desktop video conference between the users at the user stations.

Figures 2A, 2B and 2C show the components of each user station 2, 4, 6, 8, 10, 12, 14 in this embodiment.

Referring to Figure 2A, a user station comprises a conventional personal computer (PC) 24, two video cameras 26, 28 and a pair of stereo headphones 30.

PC 24 comprises a unit 32 containing, in a conventional manner, one or more

processors, memory, and sound card etc, together with a display device 34, and user input devices, which, in this embodiment, comprise a keyboard 36 and mouse 38.

PC 24 is programmed to operate in accordance with programming instructions input for example as data stored on a data storage medium, such as disk 40, and/or as a signal input to PC 24 for example over a datalink (not shown) such as the Internet, and/or entered by a user via keyboard 36.

PC 24 is connected to the Internet 20 via a connection (not shown) enabling it to transmit data to, and receive data from, the other user stations.

Video cameras 26 and 28 are provided to record video images of user 44 , and, in this embodiment, are of conventional charge coupled device (CCD) design. As will be described below, image data recorded by cameras 26 and 28 is processed by PC 24 to generate data defining the movements of user 44, and this data is then transmitted to the other user stations. Each user station stores a three-dimensional computer model of the video conference containing an avatar for each participant, and each avatar is animated in response to the data received from the user station of the corresponding participant.

In the example shown in Figure 2A, cameras 26 and 28 are positioned on top of monitor 34, but can, however, be positioned elsewhere to view user 44.

Referring to Figures 2A and 2B, a plurality of coloured markers 70, 72 are provided to be attached to the clothing of user 44. The markers each have a

different colour, and, as will be explained later, are used to determine the position of the user's torso and arms during the video conference. The markers 70 are provided on elasticated bands to be worn around the user's wrists, elbows and shoulders. A plurality of markers 70 are provided on each elasticated band so that at least one marker will be visible for each position and orientation of the user's arms. The markers 72 are provided with a suitable adhesive so that they can be removably attached to the torso of user 44, for example along a central line, as shown in Figure 2B, such as at the positions of buttons on the user's clothes.

Referring to Figure 2C, headset 30 comprises earphones 48, 50 and a microphone 52 provided on a headband 54 in a conventional manner. In addition, light emitting diodes (LEDs) 56, 58, 60, 62 and 64 are also provided on headband 54. Each of the LEDs 56, 58, 60, 62 and 64 has a different colour, and, in use, is continuously illuminated. As will be explained later, the LEDs are used to determine the position of the user's head during the video conference.

LED 56 is mounted so that it is central with respect to earphone 48 and LED 64 is mounted so that it is central with respect to earphone 50. The distance "a" between LED 56 and the inner surface of earphone 48 and between LED 64 and the inner surface of earphone 50 is pre-stored in PC 24 for use in processing to be performed during the video conference, as will be described below. LEDs 58 and 62 are slidably mounted on headband 54 so that their positions can be individually changed by user 44. LED 60 is mounted on a member 66 so that it protrudes above the top of headband 54.

In this way, when mounted on the head of user 44, LED 60 is held clear of the user's hair. Each of the LEDs 56, 58, 60, 62 and 64 is mounted centrally with respect to the width of headband 54, so that the LEDs lie in a plane defined by the headband 54.

Signals from microphone 52 and signals to headphones 48, 50 are carried to and from PC 24 via wires in cable 68. Power to LEDs 56, 58, 60, 62 and 64 is also carried by wires in cable 68.

Figure 3 schematically shows the functional units into which the components of PC 24 effectively become configured when programmed by programming instructions. The units and interconnections shown in Figure 3 are notional and are shown for illustration purposes only to assist understanding; they do not necessarily represent the exact units and connections into which the processor, memory, etc of PC 24 become configured.

Referring to Figure 3, central controller 100 processes inputs from user input devices such as keyboard 36 and mouse 38, and also provides control and processing for a number of the other functional units. Memory 102 is provided for use by central controller 100.

Image data processor 104 receives frames of image data recorded by video cameras 26 and 28. The operation of cameras 26 and 28 is synchronised so that images taken by the cameras at the same time can be processed by image data processor 104. Image data processor 104 processes synchronous frames of image data (one from camera 26 and one from camera 28) to generate data

defining (i) image pixel data for the user's face, (ii) the 3D coordinates of each of the markers 70 and 72 on the user's arms and torso, and (iii) a view parameter which, as will be explained further below, defines the direction in which the user is looking. Memory 106 is provided for use by image data processor 104.

The data output by image data processor 104 and the sound from microphone 52 is encoded by MPEG 4 encoder 108 and output to the other user stations via input/output interface 110 as an MPEG 4 bitstream.

Corresponding MPEG 4 bitstreams are received from each of the other user stations and input via input/output interface 110. Each of the bitstreams (bitstream 1, bitstream 2 .... bitstream "n") is decoded by MPEG 4 decoder 112.

Three-dimensional avatars (computer models) of each of the other participants in the video conference and a three-dimensional computer model of the conference room are stored in avatar and 3D conference model store 114.

In response to the information in the MPEG 4 bitstreams from the other participants, model processor 116 animates the stored avatars so that the movements of each avatar mimic the movements of the corresponding participant in the video conference.

Image renderer 118 renders an image of the 3D model of the conference room and the avatars, and the resulting pixel data is written to frame buffer 120 and

displayed on monitor 34 at a video rate. In this way, images of the avatars and 3D conference model are displayed to the user, and the images show the movement of each avatar corresponding to the movements of the participants in real-life.

Sound data from the MPEG 4 bitstreams received from the other participants is processed by sound generator 122 together with information from image data processor 104 defining the current position and orientation of the head of user 44, to generate signals which are output to earphones 48 and 50 in order to generate sound to user 44. In addition, signals from microphone 52 are processed by sound generator 22 so that sound from the user's own microphone 52 is heard by the user via his headphones 48 and 50.

Figure 4 shows, at a top level, the processing operations carried out to conduct a video conference between the participants at user stations 2, 4, 6, 8, 10, 12 and 14.

Referring to Figure 4, at step S2, suitable communication connections between each of the user stations 2, 4, 6, 8, 10, 12, 14 are established in a conventional manner.

At step S4, processing operations are performed to set up the video conference. These operations are performed by one of the user stations, previously designated as the conference coordinator.

Figure 5 shows the processing operations performed at step S4 to set up the

conference.

Referring to Figure 5, at step S20, the conference coordinator requests the name of each participant, and stores the replies when they are received.

At step S22, the conference coordinator requests the avatar of each participant, and stores the avatars when they are received. Each avatar comprises a three-dimensional computer model of the participant, and may be provided by prior laser scanning of the participant in a conventional manner, or in other conventional ways, for example as described in University of Surrey Technical Report CVSSP - hilton98a, University of Surrey, Guildford, UK.

At step S24, the conference coordinator defines a seating plan for the participants taking part in the video conference. In this embodiment, this step comprises assigning a number to each participant (including the conference coordinator) and defining the order of the participants in a circle, for example as shown in Figure 6.

At step S26, the conference room coordinator selects whether a circular or rectangular conference room table is to be used for the video conference.

At step S28, the conference coordinator sends data via Internet 20 defining each of the avatars received at step S22 (including his own), the participant numbers and seating plan defined at step S24, the table shape selected at step S26, and the participants names received at step S20 (including his own) to each of the other participants in the video conference.

Referring again to Figure 4, at step S6, processing operations are performed to calibrate each user station 2, 4, 6, 8, 10, 12, 14 (including the user station of the conference coordinator).

Figure 7 shows the processing operations performed at step S6 to calibrate one of the user stations. These processing operations are performed at every user station.

Referring to Figure 7, at step S40, the data transmitted by the conference coordinator at step S28 (Figure 5) is received and stored. The three-dimensional avatar model of each participant is stored in its own local reference system in avatar and 3D conference model store 114. The other data is stored for example in memory 102 for subsequent use.

At step S42, central controller 100 requests user 44 to input information about the cameras 26, 28. Central controller 100 does this by displaying a message on monitor 34 requesting the user to input for each camera the focal length of the lens in millimetres and the size of the imaging charge couple device (CCD) within the camera. This may be done by displaying on monitor 34 a list of conventional cameras, for which the desired information is pre-stored in memory 102, and from which user 44 can select the camera used, or by the user inputting the information directly. At step S44, the camera parameters input by the user are stored, for example in memory 102 for future use.

At step S46, central controller 100 displays a message on monitor 34 requesting user 44 to input the width in millimetres of the screen of

monitor 34, and at step S48, the width which is input by the user is stored, for example in memory 102, for future use.

At step S49, central controller 100 displays a message on monitor 34 instructing the user to wear the headset 30 and body markers 70, 72, as previously described with reference to Figures 2A, 2B and 2C. When the user has completed this step, he inputs a signal to central controller 100 using keyboard 36. Power is then supplied to headset 30 worn by user 44 so that each of the LEDs 56, 58, 60, 62 and 64 are continuously illuminated.

At step S50, central controller 100 displays a message on monitor 34 instructing the user to position the movable LEDs 58, 62, on headset 30 so that the LEDs align with the user's eyes. When the user has slid LEDs 58 and 62 on headband 54 so that they align with his eyes, he inputs a signal to central controller 100 using keyboard 36.

At step S52, central controller 100 displays a message on monitor 34 instructing the user to position cameras 26 and 28 so that both cameras have a field of view which covers the user's position in front of PC 24. When the user has positioned the cameras, he inputs a signal to central controller 100 using keyboard 36.

At step S54, central controller 100 displays a message on monitor 34 instructing the user to move backwards, forwards, and to each side over the full range of distances that the user is likely to move during the video conference. At step S56, as the user moves, frames of image data are recorded

by cameras 26 and 28 and displayed on monitor 34, so that the user can check whether he is visible to each camera at all positions.

At step S58, central controller 100 displays a message on monitor 34 asking the user whether it is necessary to adjust the positions of the cameras so that the user is visible throughout the full range of his likely movements. If the user inputs a signal using keyboard 36 indicating that camera adjustment is necessary, steps S52 to S58 are repeated until the cameras are correctly positioned. On the other hand, if the user inputs a signal indicating that the cameras are correctly positioned, then processing proceeds to step S60.

At step S60, central controller 100 processes the data defining the avatar of user 44 to determine the user's head ratio, that is, the ratio of the width of the user's head (defined by the distance between the user's ears) and the length of the user's head (defined by the distance between the top of the user's head and the top of his neck), and also the width of the user's head in real-life (which can be determined since the scale of the avatar is known). The head ratio and real-life width are stored, for example in memory 106 for subsequent use by the image data processor 104.

At step S62, central controller 100 and image data processor 104 use the frames of image data previously recorded at step S56 (after the cameras 26 and 28 had been positioned for the final time) to determine the camera transformation model to be used during the video conference. The camera transformation model defines the relationship between the image plane (that is, the plane of the CCD) of camera 26 and the image plane of camera 28

which will be used to reconstruct the three-dimensional positions of the headset LEDs 56, 58, 60, 62, 64 and the body markers 70, 72 using images of these LEDs and markers recorded by the cameras 26 and 28.

Figure 8 shows the processing operations performed by central controller 100 and image data processor 104 at step S62 to determine the camera transformation model.

Referring to Figure 8, at step S90, the frames of image data recorded at step S56 are processed to identify the pair of synchronous images (that is, the image from camera 26 and the image from camera 28 recorded at the same time) which show the most left position, the pair which show the most right position, the pair which show the most forward position, and the pair which show the most backward position to which the user moved. In this embodiment, step S90 is performed by displaying the sequence of images recorded by one of the cameras at step S56, and instructing the user to input a signal, for example via keyboard 36 or mouse 38, when the image for each of the extreme positions is displayed. As noted above, these positions represent the extents of the user's likely movement during the video conference. As well as images for the most forward and backward positions, images for the most left position and most right position are identified and considered in subsequent processing to determine the camera transformation model since each of the cameras 26 and 28 is positioned at an angle to the user, and so movement of the user to the right or left increases or decreases the distance of the user from each of the cameras.

At step S92, the image data for each of the four pairs of images identified at step S90 (that is, the pair of images for the most left position, the pair of images for the most right position, the pair of images for the most forward position and the pair of images for the most backward position) is processed to identify the positions of the LEDs 56, 58, 60, 62, 64 and coloured body markers 70, 72 which are visible in each image of the pair and to match each of the identified points between the images in the pair. In this step, since each LED and each body marker has a unique predetermined colour, the pixel data for each image in a synchronised pair is processed to identify those pixels having one of the predetermined colours by examining the RGB values of the pixels. Each group of pixels having one of the predetermined colours is then processed using a convolution mask to find the coordinates within the image as a whole of the centre of the group of pixels. This is performed in a conventional manner, for example as described in "Affine Analysis of Image Sequences" by L.S. Shapiro, Cambridge University Press, 1995, ISBN 0-521-55063-7, pages 16-23. The matching of points between images is done by identifying the point in each image which has the same colour (of course, if a marker or LED is visible to only one of the cameras 26 or 28, and hence appears in only one image, then no matched pair of points will be identified for this LED or marker).

At step S94, the coordinates of the matched points identified at step S92 are normalised. Up to this point, the coordinates of the points are defined in terms of the number of pixels across and down an image from the top left hand corner of the image. At step S94, the camera focal length and image plane size previously stored at step S44 are used to convert the coordinates of the points

from pixels to a coordinate system in millimetres having an origin at the camera optical centre. The millimetre coordinates are related to the pixel coordinates as follows:

$$x* = h \times (x - C_x) \quad \text{....(1)}$$

$$y* = -v \times (y - C_y) \quad \text{....(2)}$$

where (x*,y*) are the millimetre coordinates, (x,y) are the pixel coordinates, (C_x,C_y) is the centre of the image (in pixels), which is defined as half of the number of pixels in the horizontal and vertical directions, and "h" and "v" are the horizontal and vertical distances between adjacent pixels (in mm).

At step S96, a set is formed of all the matched pairs of points identified at step S92. This combined set therefore contains points for all four pairs of images. Of course, the number of points in the combined set from each pair of images may be different, depending upon which LEDs and body markers are visible in the images. However the large number of body markers and LEDs ensures that at least seven markers or LEDs will be visible in each image, giving a minimum of $4 \times 7 = 28$ pairs of matched points in the combined set.

At step S98, a measurement matrix, M, is set up as follows for the points in the combined set created at step S96:

$$M = \begin{pmatrix} x_1x_1' & -y_1x_1' & x_1' & -x_1y_1' & y_1y_1' & -y_1' & x_1 & -y_1 & 1 \\ x_2x_2' & -y_2x_2' & x_2' & -x_2y_2' & y_2y_2' & -y_2' & x_2 & -y_2 & 1 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ x_kx_k' & -y_kx_k' & x_k' & -x_ky_k' & y_ky_k' & -y_k' & x_k & -y_k & 1 \end{pmatrix} \quad ...(3)$$

where (x,y) are the pixel coordinates of the point in the first image of a pair, (x',y') are the pixel coordinates of the corresponding (matched) point in the second image of the pair, and the numbers 1 to k indicate to which pair of points the coordinates correspond (there being k pairs of points in total).

At step S100, the most accurate camera transformation for the matched points in the combined set is calculated. By calculating this transformation using the combined set of points created at step S96, the transformation is calculated using points matched in a pair of images representing the user's most left position, a pair of images representing the user's most right position, a pair of images representing the user's most forward position, and a pair of images representing the user's most backward position. Accordingly, the calculated transformation will be valid over the user's entire workspace.

Figure 9 shows the processing operations performed at step S100 to calculate the most accurate camera transformation.

Referring to Figure 9, at step S130, a perspective transformation is calculated, tested and stored.

Figure 10 shows the processing operations performed at step S130.

Referring to Figure 10, at step S140, the next seven pairs of matched points in the combined set created at step S96 are selected (this being the first seven pairs the first time step S140 is performed).

At step S142, the selected seven pairs of points and the measurement matrix set at step S98 are used to calculate the fundamental matrix, F, representing the geometrical relationship between the cameras, F being a three by three matrix satisfying the following equation:

$$(x' \; y' \; 1) \; F \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0 \qquad \ldots.(4)$$

where (x,y,1) are the homogeneous pixel coordinates of any of the seven selected points in the first image of a pair, and (x',y',1) are the corresponding homogeneous pixel coordinates in the second image of the pair.

The fundamental matrix is calculated in a conventional manner, for example using the technique disclosed in "Robust Detection of Degenerate Configurations Whilst Estimating the Fundamental Matrix" by P.H.S. Torr, A. Zisserman and S. Maybank, Oxford University Technical Report 2090/96.

It is possible to select more than seven pairs of matched points at step S140

and to use these to calculate the fundamental matrix at step S142. However, seven pairs of points are used in this embodiment, since this has been shown empirically to produce satisfactory results, and also represents the minimum number of pairs needed to calculate the parameters of the fundamental matrix, reducing processing requirements.

At step S144, the fundamental matrix, F, is converted into a physical fundamental matrix, F_{phys}, using the camera data stored at step S44 (Figure 7). This is again performed in a conventional manner, for example as described in "Motion and Structure from Two Perspective Views: Algorithms, Error Analysis and Error Estimation" by J. Weng, T.S. Huang and N. Ahuja, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, No. 5, May 1989, pages 451-476, and as summarised below.

First the essential matrix, E, which satisfies the following equation is calculated:

$$(x*' \; y*' \; f) \; E \begin{pmatrix} x* \\ y* \\ f \end{pmatrix} = 0 \qquad \ldots.(5)$$

where (x*, y*, f) are the coordinates of any of the selected seven points in the first image in a millimetre coordinate system whose origin is at the centre of the image, the z coordinate having being normalised to correspond to the focal length, f, of the camera, and (x*', y*', f) are the corresponding coordinates of the matched point in the second image of the pair. The fundamental matrix, F, is converted into the essential matrix, E, using the following equations:

$$A = \begin{pmatrix} 1/h & 0 & c_x/f \\ 0 & 1/v & -c_y/f \\ 0 & 0 & 1/f \end{pmatrix} \quad \ldots.(6)$$

$$M = A^T F A \quad \ldots.(7)$$

$$E = \sqrt{\frac{2}{tr(M^T M)}} \times M \quad \ldots.(8)$$

where the camera parameters "h", "v", "c_x", "c_y" and "f" are as defined previously, the symbol T denotes the matrix transpose, and the symbol "tr" denotes the matrix trace.

The calculated essential matrix, E, is then converted into a physical essential matrix, "E_{phys}", by finding the closest matrix to E which is decomposable directly into a translation vector (of unit length) and rotation matrix (this closest matrix being E_{phys}).

Finally, the physical essential matrix is converted into a physical fundamental matrix, using the equation:

$$F_{phys} = A^{-1T} E_{phys} A^{-1} \quad \ldots.(9)$$

where the symbol "-1" denotes the matrix inverse.

Each of the physical essential matrix, E_{phys}, and the physical fundamental matrix, F_{phys} is a "physically realisable matrix", that is, it is directly decomposable into a rotation matrix and translation vector.

The physical fundamental matrix, F_{phys}, defines a curved surface in a four-dimensional space, represented by the coordinates (x, y, x', y') which are known as "concatenated image coordinates". The curved surface is given by Equation (4) above, which defines a 3D quadric in the 4D space of concatenated image coordinates.

At step S146, the calculated physical fundamental matrix is tested against each pair of points that were used to calculate the fundamental matrix at step S142. This is done by calculating an approximation to the 4D Euclidean distance (in the concatenated image coordinates) of the 4D point representing each pair of points from the surface representing the physical fundamental matrix. This distance is known as the "Sampson distance", and is calculated in a conventional manner, for example as described in "Robust Detection of Degenerate Configurations Whilst Estimating the Fundamental Matrix" by P.H.S. Torr, A. Zisserman and S. Maybank, Oxford University Technical Report 2090/96.

Figure 11 shows the processing operations performed at step S146 to test the physical fundamental matrix.

Referring to Figure 11, at step S170, a counter is set to zero. At step S172, the tangent plane of the surface representing the physical fundamental matrix at the four-dimensional point defined by the coordinates of the next pair of points in the seven pairs of points (the two coordinates defining each point in the pair being used to define a single point in the four-dimensional space of the concatenated image coordinates) is calculated. Step S172 effectively

comprises shifting the surface to touch the point defined by the coordinates of the pair of points, and calculating the tangent plane at that point. This is performed in a conventional manner, for example as described in "Robust Detection of Degenerate Configurations Whilst Estimating the Fundamental Matrix" by P.H.S. Torr, A. Zisserman and S. Maybank, Oxford University Technical Report 2090/96.

At step S174, the normal to the tangent plane determined at step S172 is calculated, and, at step S176, the distance along the normal from the point in the 4D space defined by the coordinates of the pair of matched points to the surface representing the physical fundamental matrix (the "Sampson distance") is calculated.

At step S178, the calculated distance is compared with a threshold which, in this embodiment, is set at 1.0 pixels. If the distance is less than the threshold, then the point lies sufficiently close to the surface, and the physical fundamental matrix is considered to accurately represent the relative positions of the cameras 26 and 28 for the particular pair of matched points being considered. Accordingly, if the distance is less than the threshold, at step S180, the counter which was initially set to zero at step S170 is incremented, the points are stored, and the distance calculated at step S176 is stored.

At step S182, it is determined whether there is another pair of points in the seven pairs of points used to calculate the fundamental matrix, and steps S172 to S182 are repeated until all such points have been processed as described above.

Referring again to Figure 10, at step S148, it is determined whether the physical fundamental matrix calculated at step S144 is sufficiently accurate to justify further processing to test it against all of the pairs of matched points in the combined set. In this embodiment, step S148 is performed by determining whether the counter value set at step S180 (indicating the number of pairs of points which have a distance less than the threshold tested at step S178, and hence are considered to be consistent with the physical fundamental matrix) is equal to 7. That is, it is determined whether the physical fundamental matrix is consistent with all of the points used to calculate the fundamental matrix from which the physical fundamental matrix was derived. If the counter is less than 7, the physical fundamental matrix is not tested further, and processing proceeds to step S152. On the other hand, if the counter value is equal to 7, at step S150, the physical fundamental matrix is tested against each other pair of matched points. This is performed in the same way as step S146 described above, with the following exceptions: (i) at step S170, the counter is set to 7 to reflect the seven pairs of points already tested at step S146 and determined to be consistent with the physical fundamental matrix, and (ii) the total error for all points stored at step S180 (including those stored during processing at step S146) is calculated, using the following equation:

$$Total\ error = \frac{\sqrt{\sum \frac{e_i^2}{p}}}{e_{th}} \qquad \ldots.(10)$$

where e_i is the distance for the "i"th pair of matched points between the 4D point represented by their coordinates and the surface representing the physical fundamental matrix calculated at step S176, this value being squared so that it is unsigned (thereby ensuring that the side of the surface representing the

physical fundamental matrix on which the point lies does not affect the result), p is the total number of points stored at step S180, and e_{th} is the distance threshold used in the comparison at step S178.

The effect of step S150 is to determine whether the physical fundamental matrix calculated at step S144 is accurate for each pair of matched points in the combined set, with the value of the counter at the end (step S180) indicating the total number of the points for which the calculated matrix is sufficiently accurate.

At step S152, it is determined whether the physical fundamental matrix tested at step S150 is more accurate than any previously calculated using the perspective calculation technique. This is done by comparing the counter value stored at step S180 in Figure 11 for the last-calculated physical fundamental matrix (this value representing the number of points for which the physical fundamental matrix is an accurate camera solution) with the corresponding counter value stored for the most accurate physical fundamental matrix previously calculated. The matrix with the highest number of points (counter value) is taken to be the most accurate. If the number of points is the same for two matrices, the total error for each matrix (calculated as described above) is compared, and the most accurate matrix is taken to be the one with the lowest error. If it is determined at step S152 that the physical fundamental matrix is more accurate than the currently stored one, then, at step S154 the previous one is discarded, and the new one is stored together with the number of points (counter value) stored at step S180 in Figure 11, the points themselves, and the total error calculated for the matrix.

At step S156, it is determined whether there is another pair of matched points which has not yet been considered, such that there is another unique set of seven pairs of matched points in the combined set to be processed. Steps S140 to S156 are repeated until each unique set of seven pairs of matched points has been processed in the manner described above.

Referring again to Figure 9, at step S132, an affine relationship for the matched points in the combined set is calculated, tested and stored.

Figure 12 shows the processing operations performed at step S132.

Referring to Figure 12, at step S200, the next four pairs of matched points are selected for processing (this being the first four pairs the first time step S200 is performed).

When performing the perspective calculations (step S130 in Figure 9), it is possible to calculate all of the components of the fundamental matrix, F. However, when the relationship between the cameras is an affine relationship, it is possible to calculate only four independent components of the fundamental matrix, these four independent components defining what is commonly known as an "affine" fundamental matrix.

Accordingly, at step S202, the four pairs of points selected at step S200 and the measurement matrix set at step S96 are used to calculate four independent components of the fundamental matrix (giving the "affine" fundamental matrix) using a technique such as that described in "Affine Analysis of Image

Sequences" by L.S. Shapiro, Section 5, Cambridge University Press 1995, ISBN 0-521-55063-7. It is possible to select more than four pairs of points at step S200 and to use these to calculate the affine fundamental matrix at step S202. However, in the present embodiment, only four pairs are selected since this has been shown empirically to produce satisfactory results, and also represents the minimum number required to calculate the components of the affine fundamental matrix, reducing processing requirements.

At step S204, the affine fundamental matrix is tested against each pair of matched points in the combined set using a technique such as that described in "Affine Analysis of Image Sequences" by L.S. Shapiro, Section 5, Cambridge University Press, 1995, ISBN 0-521-55063-7. The affine fundamental matrix represents a flat surface (hyperplane) in four-dimensional, concatenated image space, and this test comprises determining the distance between a point in the four-dimensional space defined by the coordinates of a pair of matched points and the flat surface representing the affine fundamental matrix. As with the tests performed during the perspective calculations at step S146 and S150 (Figure 10), the test performed at step S204 generates a value for the number of pairs of points for which the affine fundamental matrix represents a sufficiently accurate solution to the camera transformation and a total error value for these points.

At step S206, it is determined whether the affine fundamental matrix calculated at step S202 and tested at step S204 is more accurate than any previously calculated. This is done by comparing the number of points for which the matrix represents an accurate solution with the number of points for

the most accurate affine fundamental matrix previously calculated. The matrix with the highest number of points is the most accurate. If the number of points is the same, the matrix with the lowest error is the most accurate. If the affine fundamental matrix is more accurate than any previously calculated, then at step S208, it is stored together with the points for which it represents a sufficiently accurate solution, the total number of these points and the matrix total error.

At step S210, it is determined whether there is another pair of matched points to be considered, such that there exists another unique set of four pairs of matched points in the combined set to be processed. Steps S200 to S210 are repeated until each unique set of four pairs of matched points are processed in the manner described above.

Referring again to Figure 9, at step S134, the most accurate transformation is selected from the perspective transformation calculated at step S130 and the affine transformation calculated at step S132. This step is performed by comparing the number of points which are consistent with the most accurate perspective transformation (stored at step S154) with the number of points which are consistent with the most accurate affine transformation (stored at step S208), and selecting the transformation which has the highest number of consistent points (or the transformation having the lowest matrix total error if the number of consistent points is the same for both transformations).

Referring again to Figure 8, at step S104, it is determined whether the affine transformation is the most accurate camera transformation.

If it is determined at step S104 that the affine transformation is not the most accurate transformation, then, at step S106, the perspective transformation which was determined at step S100 is selected for use during the video conference. Subsequently, at step S108, the physical fundamental matrix for the perspective transformation is converted to a camera rotation matrix and translation vector. This conversion is performed in a conventional manner, for example as described in the above-referenced "Motion and Structure from Two Perspective Views: Algorithms, Error Analysis and Error Estimation" by J. Weng, T.S. Huang and N. Ahuja, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 5, May 1989, pages 451-476.

In the processing described above with respect to Figure 10, a fundamental matrix is calculated (steps S142) and converted to a physical fundamental matrix (step S144) for testing against the matched points (steps S146 and S150). This has the advantage that, although additional processing is required to convert the fundamental matrix to a physical fundamental matrix, the physical fundamental matrix ultimately converted at step S108 has itself been tested. If the fundamental matrix was tested, this would then have to be converted to a physical fundamental matrix which would not, itself, have been tested.

On the other hand, if it is determined at step S104, that the affine transformation is the most accurate transformation, then, at step S110, the affine transformation is selected for use during the video conference.

At step S112, the affine fundamental matrix is converted into three physical

variables describing the camera transformation, namely the magnification, "m", of the object between images recorded by the cameras, the axis, ϕ, of rotation of the camera, and the cyclotorsion rotation, θ, of the camera. The conversion of the affine fundamental matrix into these physical variables is performed in a conventional manner, for example as described in "Affine Analysis of Image Sequences" by L.S. Shapiro, Cambridge University Press, 1995, ISBN 0-521-55063-7, Section 7.

Referring again to Figure 7, at step S64, the position of the headset LEDs 56, 58, 60, 62 and 64 relative to the head of user 44 is determined. This step is performed since this relative position will depend on how the user has placed the headset 30 on his head. More particularly, as illustrated in Figure 13, the plane 130 in which the headset LEDs lie is determined by the angle at which the user wears the headset 30. Accordingly, the plane 130 of the headset LEDs may be different to the actual plane 132 of the user's head. At step S64, therefore, processing is carried out to determine the angle θ between the plane 130 of the headset LEDs and the actual plane 132 of the user's head.

Figure 14 shows the processing operations performed at step S64.

Referring to Figure 14, at step S230, central controller 100 displays a message on monitor 34 instructing the user 44 to look directly at the camera to his right (that is, camera 28 in this embodiment).

At step S232, a frame of image data is recorded with both camera 26 and camera 28 while the user is looking directly at camera 28.

At step S234, the synchronous frames of image data recorded at step S232 are processed to calculate the 3D positions of the headset LEDs 56, 58, 60, 62 and 64.

Figure 15 shows the processing operations performed at step S324 to calculate the 3D positions of the headset LEDs.

Referring to Figure 15, at step S250, the position of each headset LED 56, 58, 60, 62 and 64 is identified in each of the images recorded at step S232. The identification of the LED positions at step S250 is carried out in the same way as previously described with respect to step S92 (Figure 8).

At step S252, the positions of the next pair of LEDs matched between the pair of images are considered (this being the first pair the first time step S252 is performed), and the camera transformation model previously determined at step S62 (Figure 7) is used to calculate the projection of a ray from the position of the LED in the first image through the optical centre of the camera for the first image, and from the position of the matched LED in the second image through the optical centre of the camera for the second image. This is illustrated in Figure 16. Referring to Figure 16, ray 140 is projected from the position of an LED (such as LED 56) in the image 142 recorded by camera 26 through the optical centre of camera 26 (not shown), and ray 144 is projected from the position of the same LED in image 146 recorded by camera 28, through the optical centre of camera 28 (not shown).

Referring again to Figure 15, at step S254, the mid-point 148 (Figure 16) of

the line segment which connects, and is perpendicular to, both of the rays projected in step S252 is calculated. The position of this mid-point represents the physical position of the LED in three dimensions.

At step S256, it is determined whether there is another one of the LEDs 56, 58, 60, 62 or 64 to be processed. Steps S252 to S256 are repeated until the three-dimensional coordinates of each of the LEDs has been calculated as described above.

Referring again to Figure 14, at step S236, the plane 130 (Figure 13) in which the three-dimensional positions of the headset LEDs lie is determined, and the angle θ between this plane and the imaging plane of the camera at which the user was looking when the frames of image data were recorded at step S232 is calculated. Since the user was looking directly at the camera to his right when the frames of image data were recorded at step S232, the direction of the imaging plane of the camera to the user's right corresponds to the direction of the plane 132 of the user's head (Figure 13). Accordingly, the angle calculated at step S236 is the angle θ between the plane 130 of the headset LEDs and the plane 132 of the user's head.

Referring again to Figure 7, at step S66, the position of the display screen of monitor 34 is determined and a coordinate system is defined relative to this position.

Figure 17 shows the processing operations performed at step S66.

Referring to Figure 17, at step S270, central controller 100 displays a message on monitor 34 instructing the user to sit centrally and parallel to the display screen of the monitor 34, and to sit upright with his torso touching the edge of the desk on which PC 24 stands. At step S272, a further message is displayed instructing the user to turn but not otherwise change the position of, his head, so that the processing in the steps which follow can be carried out on the basis of a constant head position but changing head angle.

At step S274, the direction of the plane of the display screen of monitor 34 is determined. In this embodiment, this is done by determining the direction of a plane parallel to the display screen.

Figure 18 shows the processing operations performed at step S274.

Referring to Figure 18, at step S300, central controller 100 displays a marker in the centre of the display screen of monitor 34, and instructs the user to look directly at the displayed marker.

At step S302, a frame of image data is recorded with both camera 26 and 28 as the user looks at the displayed marker in the centre of the screen of monitor 34.

At step S304, the three-dimensional positions of the coloured markers 72 on the user's torso are determined. This step is carried out in the same way as step S234 in Figure 14, which was described above with respect to Figures 15 and 16, the only difference being that, since the positions of the coloured

markers 72 in each image are determined (rather than the positions of the headset LEDs), rays are projected from the positions of matched markers in each of the synchronised images. Accordingly, these steps will not be described again here.

At step S306, the three-dimensional positions of the user's headset LEDs are calculated. This step is also carried out in the same way as step S234 in Figure 14, described above with respect to Figures 15 and 16.

At step S308, the plane in which the three-dimensional positions of the headset LEDs (determined at step S306) lie is calculated.

At step S310, the direction of the plane determined at step S308 is adjusted by the angle θ determined at step S64 (Figure 7) between the plane of the headset LEDs and the plane of the user's head. The resulting direction is the direction of a plane parallel to the plane of the display screen, since the plane of the user's head will be parallel to the display screen when the user is looking directly at the marker in the centre of the screen.

Referring again to Figure 17, at step S276, the position in three dimensions of the plane of the display screen of monitor 34 is determined.

Figure 19 shows the processing operations performed at step S276.

Referring to Figure 19, at step S320, central controller 100 displays a marker in the centre of the right edge of the display screen of monitor 34, and displays

a message instructing the user to look at the marker.

At step S322, a frame of image data is recorded with both camera 26 and 28 as the user looks at the marker displayed at the edge of the display screen.

At step S324, the angle of the user's head relative to the display screen about a vertical axis is determined.

Figure 20 shows the processing operations performed at step S324.

Referring to Figure 20, at step S340, the three-dimensional positions of the headset LEDs are calculated. This step is carried out in the same manner as step S234 in Figure 14, and described above with respect to Figures 15 and 16. Accordingly, the processing operations will not be described again here.

At step S342, the plane which passes through the three-dimensional positions of the headset LEDs is determined, and, at step S344, the position of this plane is adjusted by the headset offset angle θ (calculated at step S64 in Figure 7) to give the plane of the user's head.

At step S346, the angle between the direction of the plane of the user's head determined at step S344 and the direction of the plane parallel to the display screen determined at step S274 (Figure 17) is calculated. This calculated angle is the angle of the user's head relative to the plane of the display screen about a vertical axis, and is illustrated in Figure 21 as angle "α".

Referring again to Figure 19, at step S326, the three-dimensional position of the display screen is calculated and stored for subsequent use. In this step, the width of the display screen previously input by the user at step S46 and stored at step S48 (Figure 7) is used together with the angle determined at step S324 of the user's head when looking at a point at the edge of the display screen to calculate the 3D position of the display screen. More particularly, referring to Figure 21, the distance "d" of the plane parallel to the display screen determined at step S274 (Figure 17) is calculated using the angle α and one half of the width "W" of the display screen, thereby determining the three-dimensional position of the plane of the display screen. The extents of the display screen in the horizontal direction are then determined using the width "W".

Referring again to Figure 17, at step S278, a three-dimensional coordinate system and scale is defined relative to the three-dimensional position of the display screen. This coordinate system will be used to define the three-dimensional position of points which are transmitted to the other participants during the video conference. Accordingly, each participant uses the same coordinate system and scale, and therefore transmits coordinates which can be interpreted by the other participants. Referring to Figure 22, in this embodiment, the coordinate system is defined with the origin at the centre of the display screen, the "x" and "y" axes lying in the plane of the display screen in horizontal and vertical directions respectively, and the "z" axis lying in a direction perpendicular to the plane of the display screen in a direction towards the user. The scale for each axis is predefined (or could, for example, be transmitted to each user station by the conference coordinator).

Also at step S278, the transformation is calculated which maps three-dimensional coordinates calculated using the camera transformation model determined at step S62 to the new, standardised coordinate system and scale. This transformation is calculated in a conventional manner, with scale changes being determined by using the width of the user's head in real-life (determined at step S60 in Figure 7) and the distance "a" between each of LEDs 56 and 64 and the inner surface of the earphones 48, 50 (Figure 2C) to determine the distance between the LEDs 56 and 64 in real-life when the headset 30 is worn by the user, and by using this real-life LED separation to relate the distance between the three-dimensional coordinates of the headset LEDs 56 and 64 calculated using the camera transformation model at step S306 in Figure 18 to the predefined scale of the standard coordinate system.

At step S280, the three-dimensional positions of the body markers 72 previously calculated at step S304 (Figure 18) are transformed into the standard coordinate system defined at step S278.

At step S282, the three-dimensional positions of the body markers 72 in the standard coordinate system are transmitted to the other participants in the video conference, for subsequent use in positioning the user's avatar in the three-dimensional computer model of the conference room, as will be described below.

Referring again to Figure 7, at step S68, a three-dimensional computer model is set up of the conference room table to be used for the video conference. In this embodiment, three-dimensional computer models are pre-stored of a

rectangular and semi-circular conference room table, and the appropriate model is selected for use in dependence upon the instructions received from the conference room coordinator at step S40 defining the shape of the conference room table to be used.

In addition, name labels showing the name of each of the participants are placed on the conference room table in the three-dimensional computer model, with the name displayed on each label being taken from the names of the participants received from the conference coordinator at step S40. In order to determine the positions for the name labels on the conference table, the seating position of each participant is first determined using the seating plan received from the conference coordinator at step S40. Although the conference coordinator defined the seating plan by defining the order of the participants in a circle (step S24 in Figure 5, and Figure 6), at step S68 the positions of the avatars around the conference room table are set so that, when an image of the avatars and conference room table is displayed to the user, the avatars are spread apart across the width of the display screen of monitor 34. In this way, each avatar occupies its own part of the display screen in the horizontal direction and all of the avatars can be seen by the user.

Figures 23A, 23B, 23C, 23D and 23E illustrate how the positions of avatars are set in this embodiment for different numbers of participants in the video conference.

Referring to Figures 23A, 23B, 23C, 23D and 23E in general, the avatars are spaced apart evenly around a semi-circle 164 in three dimensions. The

diameter of the semi-circle 164 (which is the same irrespective of the number of participants in the video conference) and the viewing position from which images are rendered for display to the user are chosen so that each avatar occupies a unique position across the display screen and the outermost avatars are close to the edges of the display screen in the horizontal direction. In this embodiment, the avatars are positioned around semi-circle 164 and a viewing position is defined such that the positions at which the avatars appear in an image are shown in the table below.

NUMBER OF AVATARS DISPLAYED	POSITION OF AVATAR IN IMAGE (W = screen width)
2	±0.46W
3	0.00W; ±0.46W
4	±0.20W; ±0.46W
5	0.00W; ±0.20W; ±0.46W
6	±0.12W; ±0.34W; ±0.46W

Table 1

Referring to Figure 23A, when there are three participants in the video conference, the avatars 160 and 162 for the two participants other than the user at the user station being described are positioned behind the same, straight edge of a conference room table at the ends of the semi-circle 164. As set out in the table above, avatar 160 is positioned so that it appears in an image at a

distance -0.46W from the centre of the display screen in a horizontal direction, and avatar 162 is positioned so that it appears at a distance +0.46W from the centre. Name plates 166 and 168 showing the respective names of the participants are placed on the conference room table in front of the avatars facing the viewing position from which images of the conference room table and avatars will be rendered. In this way, the user, when viewing the display, can read the name of each participant.

Figure 23B shows an example in which there are four participants of the video conference and a rectangular conference room table has been selected by the conference organiser. Again, the avatars 170, 172 and 174 for the three participants other than the user at the user station are arranged around the semi-circle 164 with equal spacing. Avatar 170 is positioned so that it appears in an image at a distance -0.46W from the centre of the display screen in a horizontal direction, avatar 172 is positioned so that it appears at the centre of the display screen (in a horizontal direction), and avatar 174 is positioned so that it appears at a distance +0.46W from the centre. A name label 176, 178, 180 is placed on the conference room table facing the viewing position from which images of the conference room table and avatars will be rendered.

Figure 23C shows an example in which there are four participants of the video conference, as in the example of Figure 23B, but the conference coordinator has selected a circular conference room table. In this case, the edge of the model of the conference room table follows the semi-circle 164.

Figure 23D shows an example in which there are seven participants in the

video conference, and a rectangular conference room table is specified by the conference coordinator. The avatars 190, 192, 194, 196, 198, 200 for each of the participants other than the user at the user station are equally spaced around semi-circle 164, such that, when an image is rendered, the avatars occupy positions of -0.46W, -0.34W, -0.12W, +0.12W, +0.34W and +0.46W respectively from the centre of the display screen in a horizontal direction. A name label 202, 204, 206, 208, 210, 212 is provided for each participant facing the viewing position from which images will be rendered so that the participants' names are visible in the image displayed on monitor 34 to the user.

The relative positions and orientations of the avatars around the conference room table will be different for the participant at each user station. Referring to the seating plan shown in Figure 6, and assuming that the user at the user station being described is participant 1, then participant 2 is to the left of the user and participant 7 is to the right of the user. Accordingly, as shown in Figure 23D, the position of avatar 190 for participant 2 is set so that it appears on the left of the image, and the position of avatar 200 for participant 7 is set so that it appears on the right of the image. The positions of avatars 192, 194, 196 and 198 for participants 3, 4, 5 and 6 respectively are arranged between the positions of avatars 190 and 200 in accordance with the order defined in the seating plan.

Similarly, by way of further example, the positions of the avatars would be set at the user station of participant 2 so that the order of the participants from left to right in an image is 3, 4, 5, 6, 7 and 1.

The example shown in Figure 23E corresponds to that shown in Figure 23D, except that a circular conference room table is specified by the conference coordinator.

Referring again to Figure 7, at step S70, a respective transformation is defined for each participant which maps the avatar for the participant from the local coordinate system in which it was stored at step S40 into the three-dimensional computer model of the conference room created at step S68 so that the avatar appears at the correct position at the conference room table. In this step, the three-dimensional positions of the body markers 72 previously received from each participant (as transmitted at step S282 in Figure 17) when the participant was sitting with his torso against the edge of his desk are used to determine the transformation such that the edge of the user's desk maps to the edge of the conference room table where the avatar is placed.

At step S72, data is stored, for example in memory 106, defining the relationship between each of the avatars which will be displayed to the user (that is, the avatars of the other participants) and the horizontal position on the display screen of monitor 34 at which the avatar will be displayed. As described above with respect to step S68, the avatars are positioned in the conference room model such that the position at which each avatar will appear across the display screen in a horizontal direction when an image is rendered is fixed. Accordingly, in this embodiment, data defining these fixed positions for each different number of participants is pre-stored in memory 106, and, at step S72, the data defining the fixed positions for the correct number of

participants is selected and each of the fixed positions is assigned a participant number (received from the conference coordinator at step S40) defining the participant displayed at that position. More particularly, as will now be described with reference to Figure 24, data defining a piece-wise linear function between the fixed positions of the avatars is stored and the participant numbers are associated with this data at step S72.

Referring to Figure 24, data for the display of six avatars is shown (corresponding to the examples described previously with respect to Figure 23D and Figure 23E). The vertical axis in Figure 24 shows horizontal screen position, and values on this axis range from -0.5 (corresponding to a position on the left hand edge of the screen) to +0.5 (corresponding to a position on the right hand edge of the screen). The horizontal axis has six equally spaced divisions 400, 402, 404, 406, 408 and 410, each of which corresponds to a participant. Accordingly, the value of the function at each of these positions on the horizontal axis is -0.46, -0.34, -0.12, +0.12, +0.34 and +0.46 respectively (as shown by the dots in Figure 24) since these are the horizontal screen positions at which the avatars for six participants will be displayed. Data is also stored defining a piece-wise linear function between each of these values. At step S72, each of the six positions on the horizontal axis is assigned a participant number corresponding to the participant whose avatar will be displayed at the associated horizontal screen position. Referring to the seating plane shown in Figure 6, in this example, position 400 is allocated participant number 2, position 402 is allocated participant number 3, position 404 is allocated participant number 4, position 406 is allocated participant number 5, position 408 is allocated participant number 6 and

position 410 is allocated participant number 7. It should be noted that the participant numbers for each of these positions will be different for each user station. By way of example, at the user station for participant 2, the participant numbers allocated to positions 400, 402, 404, 406, 408 and 410 will be 3, 4, 5, 6, 7 and 1 respectively.

As a result of allocating the participant numbers, the piece-wise linear function therefore defines, for each horizontal screen position a so-called "view parameter" V for the user which defines which participant in the conference room the user is looking at when he is looking at a particular position on the display screen of monitor 34. As will be explained below, during the video conference, processing is carried out to determine the horizontal position on the display screen which the user is looking, and this is used to read the "view parameter" V for the user, which is then transmitted to the other participants to control the user's avatar.

Referring again to Figure 7, at step S74, when all of the preceding steps in Figure 7 have been completed, a "ready" signal is transmitted to the conference coordinator indicating that the user station has been calibrated and is now ready to start the video conference.

Referring again to Figure 4, at step S8, the video conference itself is carried out.

Figure 25 shows the processing operations which are performed to carry out the video conference.

Referring to Figure 25, the processes at steps S370, S372, S374-1 to S374-6, S376 and S378 are carried out simultaneously.

At step S370, frames of image data are recorded by cameras 26 and 28 as the user participates in the video conference, that is as the user views the images of the avatars of the other participants on monitor 34, listens to the sound data from the other participants and speaks into microphone 52. Synchronous frames of image data (that is, one frame from each camera which were recorded at the same time) are processed by image data processor 104 at video frame rate to generate in real time data defining the three-dimensional coordinates of the body markers 70, 72, the view parameter V defining where the user was looking in the conference room when the images were recorded, and pixel data for the face of the user. This data is then transmitted to all of the other participants. Step S370 is repeated for subsequent pairs of frames of image data until the video conference ends.

Figure 26 shows the processing operations performed at step S370 for a given pair of synchronised frames of image data.

Referring to Figure 26, at step S390, synchronous frames of image data are processed to calculate the three-dimensional coordinates of the headset LEDs 56, 58, 60, 62, 64 and body markers 70, 72 which are visible in both of the images. This step is carried out in the same way as step S234 in Figure 14, and described above with respect to Figures 15 and 16, except that the processing is performed for the body markers 70, 72 in addition to the headset LEDs. Accordingly, this processing will not be described again here.

At step S392, the plane of the user's head is determined by finding the plane which passes through the three-dimensional positions of the headset LEDs calculated at step S390 and adjusting this plane by the headset offset angle θ previously determined at step S64 (Figure 7).

At step S394, a line is projected from the plane of the user's head in a direction perpendicular to this plane, and the intersection of the projected line with the display screen of monitor 34 is calculated. This is illustrated in Figures 27A, 27B and 27C.

Referring to Figure 27A, in this embodiment, the mid-point 220 of the line between the three-dimensional coordinates of the headset LEDs 58 and 62 is determined and a line 218 is projected from the calculated mid-point 220 perpendicular to the plane 224 of the user's head (which was calculated at step S392 by determining the plane 228 of the headset LEDs and adjusting this by the headset offset angle θ). As described above with respect to step S50 (Figure 7), the headset LEDs 58 and 62 are aligned with the user's eyes so that, in this embodiment, the projected line 218 is not only perpendicular to the plane 224 of the user's head, but also passes through a point on this plane representative of the position of the user's eyes.

Referring to Figure 27B, the projected line 218 intersects the plane of the display screen of monitor 34 at a point 240. In step S394, the horizontal distance "h" shown in Figure 27C of the point 240 from the centre of the display screen (that is, the distance between the vertical line in the plane of the display screen on which point 240 lies and the vertical line in the plane of the

display screen on which the centre point of the display lies) is calculated using the three-dimensional coordinates of the display screen previously determined at step S66 (Figure 7) during calibration.

Referring again to Figure 26, at step S396, the view parameter V defining where the user was looking when the frames of image data being processed were recorded is determined. More particularly, the ratio of the distance "h" calculated at step S394 to the width "W" of the display screen stored at step S48 (Figure 7) is calculated and the resulting value is used to read a value for the view parameter V from the data stored at step S72 during calibration. By way of example, if the distance "h" is calculated to be 2.76 inches and the width "W" of the display screen is 12 inches (corresponding to a 15 inch monitor), then a ratio of 0.23 would be calculated and, referring to Figure 24, this would cause a view parameter "V" of 5.5 to be generated. As can be seen from the example shown in Figures 27B and 27C, the projected ray 218 indicates that the user 44 is looking between participants 5 and 6, and hence a view parameter of 5.5 would define this position.

Referring again to Figure 26, at step S398, the direction of the imaging plane of each of the cameras 26 and 28 (that is, the plane in which the CCD of the camera lies) is compared with the direction of the plane of the user's head calculated at step S392 to determine which camera has an imaging plane most parallel to the plane of the user's head. Referring again to Figure 27B, for the example illustrated, it will be seen that the imaging plane 250 for camera 28 is more parallel to the plane 224 of the user's head than the imaging plane 252 of camera 26. Accordingly, in the example illustrated in Figure 27B, camera

28 would be selected at step S398.

At step S400, the frame of image data from the camera selected at step S398 is processed to extract the pixel data representing the user's face in the image. In this embodiment, this step is performed using the three-dimensional positions of the headset LEDs 56 and 64 calculated at step S390, the size and ratio of the user's head determined at step S60 (Figure 7) and the distance "a" between each LED 56, 64 and the inner surface of the corresponding earpiece 48, 50 (which, as noted above, is pre-stored in PC 24). More particularly, using the three-dimensional positions of the headset LEDs 56 and 64, and the distance "a", the points representing the extents of the width of the user's head in three dimensions are determined. These extent points are then projected back into the image plane of the camera selected at step S398 using the camera transformation determined at step S62 (Figure 7). The projected points represent the extents of the width of the user's head in the image, and, using the value of this width and the ratio of the user's head length, the extents of the user's head length in the image are determined. Pixels representing the image between the extents of the width of the user's head and the extents of the length of the user's head are then extracted. In this way, image data is not extracted which shows the headset 30 which the user is wearing.

At step S401, the three-dimensional coordinates of the body markers 70, 72 calculated at step S390 are transformed into the standardised coordinate system previously defined at step S66 in Figure 7.

At step S402, MPEG 4 encoder 108 encodes the face pixel data extracted at

step S400, the 3D coordinates of the body markers 70, 72 generated at step S401 and the view parameter determined at step S396 in accordance with the MPEG 4 standard. More particularly, the face pixel data and the 3D coordinates are encoded as a Movie Texture and Body Animation Parameter (BAP) set and, since the MPEG 4 standard does not directly provide for the encoding of a view parameter, this is encoded in a general user data field. The encoded MPEG 4 data is then transmitted to the user stations of each of the other participants via input/output interface 110 and the Internet 20.

Referring again to Figure 25, at step S372, sound produced by user 44 is recorded with microphone 52 and encoded by MPEG 4 encoder 108 in accordance with the MPEG 4 standard. The encoded sound is then transmitted to the other participants by input/output interface 110 and the Internet 20.

At steps S374-1 to S374-6, MPEG decoder 112, model processor 116 and central controller 100 perform processing to change the avatar models stored in avatar and 3D conference model store 114 in dependence upon the MPEG 4 encoded data received from the other participants. More particularly, in step S374-1 processing is performed to change the avatar of the first external participant using the data received from that participant, in step S374-2 the avatar of the second external participant is changed using data received from the second external participant etc. Steps S374-1 to S374-6 are performed simultaneously, in parallel.

Figure 28 shows the processing operations performed in each of steps S374-1 to S374-6.

Referring to Figure 28, at step S420, MPEG 4 decoder 112 awaits further data from the participant whose avatar is to be updated. When data is received, it is decoded by the MPEG 4 decoder, and the decoded data is then passed to model processor 116 at step S422, where it is read to control subsequent processing by model processor 116 and central controller 100.

At step S424, the position of the avatar body and arms are changed in the three-dimensional coordinate system in which it is stored in avatar and 3D conference model store 114 so that the body and arms of the avatar fit the received three-dimensional coordinates of the body markers 70, 72 of the actual participant. In this way, the pose of the avatar is made to correspond to the real-life pose of the actual participant which the avatar represents.

At step S426, the face pixel data in the bitstream received from the participant is texture mapped onto the face of the avatar model in three dimensions.

At step S428, the avatar is transformed from the local coordinate system in which it is stored into the three-dimensional model of the conference room using the transformation previously defined at step S70 (Figure 7).

At step S430, the head of the transformed avatar in the three-dimensional conference room model is changed in dependence upon the view parameter, V, of the participant defined in the received bitstream. More particularly, the head of the avatar is moved in three dimensions so that the avatar is looking at the position defined by the view parameter. For example, if the view parameter, V, is 5, then the avatar's head is moved so that the avatar is looking

at the position in the three-dimensional conference room at which participant 5 is seated. Similarly, if, for example, the view parameter is 5.5, then the avatar's head is rotated so that the avatar is looking mid-way between the positions at which the fifth and sixth participants sit in the three-dimensional conference room.

Figures 29A, 29B and 29C illustrate how the position of the avatar's head is changed in the conference room model in dependence upon changes of the participant's head in real-life.

Referring to Figure 29A, an example is shown in which participant 1 in real-life is initially looking at participant 2 (or more particularly, the avatar of participant 2) on the display screen of his monitor, and then rotates his head through an angle $\beta 1$ to look at participant 7 on the display screen. In real-life, the angle of rotation $\beta 1$ would be approximately 20°-30° for typical screen sizes and seating positions from the screen.

Figure 29B represents the images seen by participant 3 of the video conference. When the head of participant 1 in real-life is looking at participant 2, then the head of the avatar 300 of participant 1 is positioned so that it, too, is looking at the avatar of participant 2 in the three-dimensional model of the conference room stored at the user station of participant 3. As the first participant rotates his head in real-life to look at participant 7, the head of the avatar 300 undergoes a corresponding rotation to look at the avatar of participant 7 in the three-dimensional conference room model. However, the angle $\beta 2$ through which the head of avatar 300 moves is not the same as

angle β1 through which the head of the first participant moves in real-life. In fact, in this example, the angle β2 is much larger than the angle β1 due to the relative positions of the avatars in the conference room model. Consequently, the motion of the heads of the avatars does not take place in the same coordinate system as that of the motion of the heads of the actual participants in real-life.

The change in angle of the head of avatar 300 will be different for each user station since the arrangement of the avatars in the three-dimensional conference room model is different at each user station. Figure 29C illustrates how the head of avatar 300 moves in the image displayed at the user station of participant 2 as participant 1 moves his head in real-life through the angle β1 to look from participant 2 to participant 7. Referring to Figure 29C, since participant 1 is originally looking at participant 2, the head of avatar 300 is originally directed towards the viewing position from which the image is rendered for display to participant 2. As participant 1 rotates his head through angle β1 in real-life, the head of avatar 300 is rotated through angle β3 so that the head is looking at the avatar of participant 7 in the three-dimensional model of the video conference room stored at the user station of participant 2. The angle β3 is different to both β1 and β2.

Referring again to Figure 25, at step S376, image renderer 118 and central controller 100 generate and display a frame of image data on monitor 34 showing the current status of the three-dimensional conference room model and the avatars therein. The processing performed at step S376 is repeated to display images at video rate, showing changes as the avatars are updated in

response to changes of the participants in real-life.

Figure 30 shows the processing operations performed at step S376.

Referring to Figure 30, at step S450, an image of the three-dimensional conference room model is rendered in a conventional manner to generate pixel data, which is stored in frame buffer 120.

At step S452, the current view parameter V determined at step S370 in Figure 25 (which occurs in parallel) is read. As noted above, this view parameter defines the position on the monitor at which the user is determined to be looking, relative to the avatars displayed.

At step S454, the image data generated and stored at step S450 is amended with data for a marker to show the position at which the user is determined to be looking in accordance with the view parameter read at step S452.

At step S456, the pixel data now stored in frame buffer 120 is output to monitor 34 to display an image on the display screen.

Figure 31 illustrates the display of markers in accordance with the users current view parameter V.

Referring to Figure 31, if for example it is determined at step S452 that the user's current view parameter is 5, then at step S454, image data for arrow 310 is added so that, when the image is displayed at step S456, the user sees

arrow 310 indicating that he is determined to be looking at participant 5 and that this is the information which will be transmitted to all of the other participants. Accordingly, if the displayed marker does not accurately indicate the user's intended viewing direction, the user can change the position of his head whilst watching the position of the marker change until the correct viewing direction is determined and transmitted to the other users.

By way of further example, if the user's view parameter is 6.5, then arrow 320 would be displayed (instead of arrow 310) indicating a position mid-way between the avatars of participants 6 and 7.

Referring again to Figure 25, at step S378, MPEG 4 decoder 112, central controller 100 and sound generator 122 perform processing to generate sound for the user's headset 30.

Figure 32 shows the processing operations performed at step S378.

Referring to Figure 32, at step S468 the input MPEG 4 bitstreams received from each participant are decoded by MPEG 4 decoder 112 to give a sound stream for each participant.

At step S470, the current head position and orientation for each avatar in the coordinate system of the three-dimensional computer model of the conference room are read, thereby determining a sound direction for the sound for each of the avatars.

At step S472, the current head position and orientation of the user (to whom the sound will be output) is read (this having being already determined at step S370 in Figure 25), thereby defining the direction for which the output sound is to be generated.

At step S474, the input sound streams decoded at step S468, the direction of each sound stream determined at step S470 and the output direction for which sound is to be generated determined at step S472 are input to the sound generator 122, where processing is carried out to generate left and right output signals for the user's headset 30. In this embodiment, the processing in sound generator 122 is performed in a conventional manner, for example such as that described in "The Science of Virtual Reality and Virtual Environments" by R.S. Kalawsky, Addison-Wesley Publishing Company, ISBN 0-201-63171-7, pages 184-187.

In the processing described above, at step S472, the user's current head position and orientation are used to determine an output direction which is subsequently used in the processing of the sound streams at step S474. In this way, the sound which is output to the headset 30 of the user changes in dependence upon the user's head position and orientation, even though the images which are displayed to the user on monitor 34 do not change as his head position and orientation change (other than the displayed marker indicating where the user is looking).

A number of modifications are possible to the embodiment of the invention described above.

For example, in the embodiment described above, the cameras 26 and 28 at each user station record images of a single user at the user station and processing is performed to determine transmission data for the single user. However, the cameras 26 and 28 may be used to record images of more than one user at each user station and processing may be carried out to generate the face pixel data, the three-dimensional coordinates of the body markers and the view parameter for each of the users at the user station, and to transmit this data to the other participants to facilitate the animation of an avatar corresponding to each one of the users.

In the embodiment above at steps S42 and S44 (Figure 7), camera parameters are input by the user. However, each of the cameras 26, 28 may be arranged to store these parameters and to pass it to PC 32 when the camera is connected to the PC.

In the embodiment above, LEDs 56, 58, 60, 62 and 64 are provided on headset 30. However, other forms of lights or identifiable markers may be provided instead.

In the embodiment described above, the headset LEDs 56, 58, 60, 62, 64 are continuously illuminated and have different colours to enable them to be identified in an image. Instead of having different colours, the LEDs could be arranged to flash at different rates to enable them to be distinguished by comparison of images over a plurality of frames, or the LEDs may have different colours and be arranged to flash at different rates.

In the embodiment above, the coloured body markers 70, 72 may be replaced by LEDs. Also, instead of using coloured markers or LEDs, the position of the user's body may be determined using sensors manufactured by Polhemus Inc., Vermont, USA, or other such sensors.

In the embodiment above, in the processing performed at step S370 (Figure 25) data for the whole of each image is processed at step S390 (Figure 26) to determine the position of each LED and each coloured body marker in the image. However, the position of each LED and each body marker may be tracked through successive frame of image data using conventional tracking techniques, such as Kalman filtering techniques, for example as described in "Affine Analysis of Image Sequences" by L.S. Shapiro, Cambridge University Press, 1995, ISBN 0-521-55063-7, pages 24-34.

In the embodiment above, at step S72 (Figure 7), data is stored defining the relationship between horizontal screen position and the view parameter V. Further, at step S396 (Figure 26), this stored data is used to calculate the view parameter to be transmitted to the other participants in dependence upon the horizontal distance between the point on the display screen at which the user is looking and the centre of the display screen. This method of determining the view parameter V is accurate when the viewing position from which the 3D model of the conference room and avatars is rendered is such that the participants are displayed to the user with their heads at substantially the same vertical height on the screen. However, errors can occur when the viewing position is such that the heads of the participants are at different heights on the display screen. To address this, it is possible to store data at step S72 defining

the relationship between the view parameter V and the distance of each avatar around the arc 164 (from any fixed point), and at step S396 to calculate the point on arc 164 which is nearest to the point on the screen at which the user is looking and use the calculated point on arc 164 to read the view parameter V which is to be transmitted to the other participants from the stored data. Further, although in the embodiment above the viewing position from which the 3D conference room model and avatars are rendered is fixed, it is possible to allow the user to vary this position. The view parameter V would then be calculated most accurately using the positions of the avatars around arc 164 as described above.

In the embodiment above, in the processing performed at step S370 (Figure 25), the user's view parameter is determined in dependence upon the orientation of the user's head. In addition, or instead, the orientation of the user's eyes may be used.

In the embodiment above, the sound from the user's own microphone 52 is fed to the user's headphones 48, 50. However, the user may be able to hear his own voice even when wearing the headphones, in which case such processing is unnecessary.

In the processing performed at step S62 (Figure 7) in the embodiment above, both a perspective camera transformation and an affine transformation are calculated and tested (steps S130 and S132 in Figure 9). However, it is possible to calculate and test just an affine transformation and, if the test reveals acceptable errors, to use the affine transformation during the video

conference, or, if the test reveals unacceptable errors, to calculate and use a perspective transformation.

In the embodiment above, the names of the participants displayed on the name plates are based on the information provided by each participant to the conference coordinator at step S20 (Figure 5). However, the names may alternatively be based on other information, such as the log-on information of each participant at a user station, the telephone number of each user station, or information provided in the data defining the avatar of each participant.

In the embodiment above, at step S400 (Figure 26), the face pixel data is extracted following processing to determine the extents of the user's head such that the extracted pixel data will not contain pixels showing the headset 30. Instead, the pixel data may be extracted from an image by simply extracting all data bounded by the positions of the LEDs 56, 60 and 64 and using the user's head ratio to determine the data to extract in the direction of the length of the user's face. Conventional image data interpolation techniques could then be used to amend the pixel data to remove the headset 30.

In the embodiment above, a view parameter V is calculated to define the position of the head of an avatar. In this way, movements of the user's head in real-life are appropriately scaled to give the correct movement of the avatar's head in the three-dimensional conference room models at the user stations of the other participants. In addition, it is also possible to perform corresponding processing for user gestures, such as when the user points, nods his head, etc. at a particular participant (avatar) on his display screen.

In the embodiment above, processing is performed by a computer using processing routines defined by programming instructions. However, some, or all, of the processing could be performed using hardware.

In the embodiment above, two cameras 26 and 28 are used at each user station to record frames of image data of the user 44. The use of two cameras enables three-dimensional position information to be obtained for the headset LEDs and body markers. However, instead, a single camera could be used together with a range finder to provide depth information. Further, a single calibrated camera could be used on its own, with depth information obtained using a standard technique, for example as described in "Computer and Robot vision, Volume 2" by R.M. Haralick and L.G. Shapiro, Addison-Wesley Publishing Company, 1993, ISBN 0-201-56943-4, pages 85-91.

Instead of using LEDs or coloured markers to determine the position of the user's head, arms and torso, conventional feature matching techniques could be used to match natural features of the user in each of the images in a pair of synchronised images. Examples of conventional techniques are given in "Fast visual tracking by temporal consensus" by A.H. Gee and R. Cipolla in Image and Vision Computing, 14(2): 105-114, 1996, in which nostrils and eyes are tracked and "Learning and Recognising Human Dynamics in Video Sequences" by C. Bregler, Proceedings IEEE Conference on Computer Vision and Pattern Recognition, June 1997, pages 568-574, in which blobs of motion and colour similarity corresponding to arms, legs and torso are tracked.

In the embodiment above, as well as including an avatar of each of the other

participants in the 3D computer model of the conference room stored at each user station, a three-dimensional computer model of one or more objects, such as a whiteboard, flip chart etc. may also be stored. The position of such an object may be defined in the seating plan defined by the conference coordinator at step S24 (Figure 5). Similarly, data defining a three-dimensional computer model of one or more characters (a person, animal etc.) which is to be animated during the video conference but whose movements are not related to the movements of one of the participants at the conference may also be stored in the three-dimensional computer model of the conference room at each user station. For example, the movements of such a character may be computer-controlled or controlled by a user.

In the embodiment above, at step S68 (Figure 7), the positions of the avatars around the conference room table are set using the values given in Table 1. However, other positions may be used. For example, the avatars may be arranged so that their horizontal positions on the display screen are given by the following equation:

$$W_n = 0.46W \cos\left(\frac{180i}{N-1}\right) \qquad \text{....(11)}$$

where: N is the number of avatars displayed on the screen

W_n is the position of the nth avatar (n = 1 ...N)

i = n-1

W is the screen width

Alternatively, rather than arranging the avatars at equally spaced positions

around a semi-circle as in the embodiment above or arranging the avatars so that their positions on the display screen are given by equation (11) above, processing apparatus 32 may perform processing to calculate a position for each avatar in the 3D computer model of the conference room such that, firstly, the minimum movement that the head of an avatar appears to undergo to switch gaze from one participant to another participant in the conference is maximised and, secondly, the avatars appear evenly spaced across a horizontal line on the display screen of monitor 34.

Accordingly, by maximising the minimum apparent head movement, it is easier for the user viewing the display to detect when an avatar has changed its gaze direction and at which of the other avatars it is now looking. Similarly, by arranging the avatars in the 3D computer model of the conference room so that they appear evenly spaced across the display screen, occlusion between the avatars and large amounts of unused display space are avoided.

Figure 33 shows processing operations that can be performed by processing apparatus 32 to calculate positions for the avatars in the 3D computer model of the conference room such that the minimum apparent head movement for the avatars is maximised and the avatars appear to the viewer to be evenly spaced across the display screen.

Referring to Figure 33, at step S500, values of parameters to be used in the processing are set.

Figure 34 shows the processing operations performed by processing apparatus 32 at step S500.

Referring to Figure 34, at step S520, the value of the number of avatars to be displayed to the user on monitor 34 is read (this being the value received from the conference coordinator at step S40 in Figure 7).

At step S522, the average distance of the user from the display screen of monitor 34 is calculated. More particularly, the average distance is calculated as the average of the user's foremost position and rearmost position determined from the image data recorded at step S56 (Figure 7) using the position of the display screen determined at step S66 (Figure 7).

At step S524, the half-width (that is, W/2 in Figure 21) is calculated based on the full screen width stored at step S48 (Figure 7).

At step S526, the average distance calculated at step S522 is converted to a multiple of the half-screen width calculated at step S524, thereby giving the average distance of the user from the display screen in terms of the screen half-width.

At step S528, a value defining the minimum size that an avatar can have when displayed on the display screen is read. More particularly, as in the example shown in Figures 23B to 23E, one or more of the avatars will be positioned in the 3D conference room model at a position further away from the viewing position from which an image of the model is rendered (the "rendering

viewpoint") than some of the other avatars. Accordingly, the avatar(s) positioned further from the rendering viewpoint will have a smaller size on the display screen than the avatars which are closer to the rendering viewpoint. The value read at step S528 therefore defines the smallest size which an avatar is allowed to take on the display screen. In this embodiment, the size is defined as a relative value, that is, the size is defined as a fraction of the size of the largest avatars which appear on the display screen. In this embodiment, the minimum value is pre-stored as 0.3 (so that the smallest avatar can not be less than 30% of the size of the largest avatar), although this value could be determined and input by a user.

At step S530, the minimum display size read at step S528 is used to calculate the maximum distance along the z-axis at which an avatar may be placed in the 3D computer model of the conference room. That is, a z-value is calculated beyond which avatars can not be placed in the 3D conference room model because their size would become too small on the display screen. More particularly, in this embodiment, the maximum z-value, Z_{max}, is calculated using the following equation:

$$Z_{max} = k\left[\frac{1}{S_{min}} - 1\right] \qquad \ldots(12)$$

where: k is the distance of the user from the display screen as a multiple of the screen half-width (calculated at step S526);

S_{min} is the minimum display size (read at step S528) which can take values $0 < S_{min} \leq 1$.

At step S532, a value defining the maximum display size which the smallest avatar in the display can take is read. As explained above with reference to step S528, one or more the avatars will be positioned in the 3D computer model of the conference room further away from the rendering viewpoint than other avatars. The value read at step S532 defines the maximum size which the avatar(s) furthest from the rendering viewpoint can take. In this embodiment, the maximum display value is defined as a relative value, that is a fraction of the size of the avatars which have the maximum size in the display (that is, the avatars closest to the rendering viewpoint). As with the minimum display value read at step S528, the maximum display value is pre-stored, but could be input by the user. In this embodiment, the value of the maximum display size is set to 1.0 so that, if necessary, all of the avatars can take the same size in the display.

At step S534, the maximum display size value read at step S532 is used to calculate the minimum distance along the z-axis at which an avatar can be placed in the 3D computer model of the conference room in order for the size of the avatar on the display not to exceed the maximum display value read at step S532. More particularly, in this embodiment, the minimum z-value, Z_{min} is calculated as follows:

$$Z_{min} = k\left[\frac{1}{S_{max}} - 1\right] \qquad \ldots(13)$$

where: k is as defined above for equation (12);

S_{max} is the maximum display size (read at step S532) which can

take values $0 < S_{max} \leq 1$.

At step S536, a value defining the z-axis resolution to be used in calculating avatar configurations is read. As will be explained further below, this resolution value defines a step size along the z-axis to be used for calculating different positions of the avatars in the 3D computer conference room model. In this embodiment, the z-axis resolution is pre-stored, and has the value 0.1. However, the z-axis resolution could be input by the user of PC 24.

Referring again to Figure 33, at step S502, all possible configurations of the avatars in the conference room 3D model are calculated subject to constraints determined by the maximum z value calculated at step S530, the minimum z value calculated at step S534 and the z-axis resolution read at step S536.

The processing performed at step S502 will be explained referring to Figures 35A and 35B by way of example, which schematically show a horizontal cross-section through the display screen 500 of display device 34, the real-world containing the user viewing the display device (labelled P1), and the 3D computer model of the conference room, for cases where 5 and 6 avatars are to be displayed respectively (the positions of avatars in the 3D computer model being labelled P2, P3, P4, P5 and P6).

In this embodiment, the positions of the two outermost avatars (defined by the seating plan received from the conference coordinator at step S40 in Figure 7) are fixed at the edges of the display screen 500 irrespective of the number of avatars to be displayed, and therefore have (x, z) coordinates (0, 1) and (0, -1)

since the position (0, 0) is defined to be at the centre of the display screen 500 and coordinate values are defined as multiples of the screen half-width.

At step S502, based on the number of avatars to be displayed on the display screen 500, horizontal rays (510, 520 and 530 in Figure 35A, and 540 and 550 in Figure 35B) are projected from the position of the user viewing the display (defined by the average distance, "k", calculated at step S526 in Figure 34) which divide the display screen 500 into equal size portions in the horizontal plane (the portions being of size 1/2 unit in the example of Figure 35A and 1/3 unit in the example of Figure 35B). The points at which these rays intersect the display screen 500 define the position on the display screen 500 at which avatars between the two extreme most avatars will appear to the user on the display. Accordingly, the avatars between the two extreme most avatars must be positioned in the 3D computer model of the conference room on the rays 510, 520, 530 or 540, 550 projected from the position of the viewer. At step S502, each possible configuration of the avatars between the two extreme most avatars is calculated.

More particularly, referring to Figure 35A first, the avatar which will have the smallest size on the display screen 500 is defined to be the avatar labelled P4, since, because this avatar is the central avatar in the seating plan, it will be the furthest avatar from the rendering viewpoint which is defined to be the position of the viewer, P1. Accordingly, P4 must lie along ray 520 between the minimum z value and the maximum z value calculated at steps S530 and S534 (Figure 34). The avatars P3 and P5 each have the same z value because these avatars are placed in the 3D computer model symmetrically. However,

the z-value of P3 and P5 can not exceed the z-value of P4 (since P4 is defined to be the avatar which has the smallest size in the display 500). At step S502, each possible configuration of the avatars P3, P4 and P5 is calculated subject to these constraints and the constraint that the minimum distance in the z direction between avatar positions is given by the z-axis resolution read at step S536 (0.01 in this embodiment).

In the example shown in Figure 35B, because the number of avatars to be displayed on display screen 500 is an even number, both of the avatars P3 and P4 will have the same size on display screen 500 and will be smallest avatars displayed. Accordingly, at step S502, each z-position of the points P3 and P4 is considered between the maximum z value and minimum z value calculated at steps S530 and S534 in increments of 0.01 (the z-axis resolution read at step S536).

Similar processing is carried out at step S502 for numbers of avatars other than the five avatars to be displayed in Figure 35A and the four avatars to be displayed in Figure 35B using the same principles described above.

At step S504, the next configuration from the configurations calculated at step S502 is selected for processing (this being the first configuration the first time step S504 is performed).

At step S506, a value is calculated representing the smallest amount of movement that the head of any of the avatars in the configuration selected at step S504 will appear to undergo when the heads of the avatars in the 3D

computer model are rotated to change gaze from one avatar to another.

The processing performed at step S506 will be described referring to Figures 36A, 36B and 36C by way of example, which schematically illustrate a configuration for five avatars P2, P3, P4, P5 and P6 which are to be displayed to the user P1 viewing the display.

Referring to Figure 36A, at step S506, the angle 600 through which the head of avatar P2 must turn to look from avatar P3 to P4 (or vice versa), the angle 610 through which the head of avatar P2 must turn to look from avatar P4 to P5, the angle 620 through which the head of avatar P2 must turn to look from avatar P5 to P6, and the angle 630 through which the head of avatar P2 must turn to look from avatar P6 to the user P1 is calculated. Further, in this embodiment, each of the calculated angles is scaled by multiplying the angle by a scale factor, S_c, given by:

$$S_c = \frac{k}{k+z} \qquad \ldots(14)$$

where: k is as defined above for equation (12);

z is the z-value of the avatar for which the head turn angle has been calculated.

By multiplying a head turn angle by the scale factor given by equation (14), the angle is converted from an angle in the 3D computer model through which the head of the avatar will turn to a value representing the movement through

which the user viewing the display screen 500 will see the head of the avatar turn.

Referring to Figure 36B, the angle 640 through which the head of avatar P3 must turn to look from avatar P4 to P5, the angle 650 through which the head of avatar P3 must turn to look from avatar P5 to P6, the angle 660 through which the head of avatar P3 must turn to look from avatar P6 to user P1, and the angle 670 through which the head of avatar P3 must turn to look from user P1 to avatar P2 are calculated and scaled by multiplying them by the scale factor S_c given by equation (14) above.

Likewise, referring to Figure 36C, the angle 680 through which the head of avatar P4 must turn to look from avatar P5 to P6, the angle 690 through which the head of avatar P4 must turn to look from avatar P6 to user P1, the angle 700 through which the head of avatar P4 must turn to look from user P1 to avatar P2, and the angle 710 through which the head of avatar P4 must turn to look from avatar P2 to avatar P3 are calculated and scaled by the scale factor S_c given by equation (14) above.

The head turn angles for avatars P5 and P6 do not need to be calculated at step S506 because these angles are the same as the head turn angles for avatars P3 and P2, respectively (due to the symmetrical positioning of the avatars in the 3D computer model). Similarly, in the case of the example shown in Figure 35B, head turn angles would be calculated at step S506 for avatars P2 and P3, but not avatars P4 and P5 since these would be the same as the head turn angles for avatars P3 and P2, respectively.

Also at step S506, the value of the movement which is the smallest value of those calculated is determined.

At step S508, a test is carried out to determine whether the smallest movement value identified at step S506 is larger than the currently stored movement value.

If it is determined at step S508 that the smallest movement value determined at step S506 is larger than the currently stored movement value (which, by default, will be the case when step S508 is performed for the first time), then, at step S510, the currently stored movement value is replaced with the smallest movement value determined at step S506 and the avatar configuration selected at step S504 is stored.

On the other hand, if it is determined at step S508 that the smallest movement value determined at step S506 is not larger than the currently stored movement value, then step S510 is omitted so that the currently stored movement value and currently store avatar configurations are retained.

At step S512, it is determined whether another of the avatar configurations calculated at step S502 remains to be processed. Steps S504 to S512 are repeated until each avatar configuration calculated at step S502 has been processed in the manner described above.

As a result of performing the processing at steps S500 to S512, the positions in the 3D computer model of the conference room have been calculated for the

avatars to be displayed which ensure that when an image of the 3D computer model is rendered from the average distance position of the viewer (P1 in Figures 35A and 35B), the avatars will appear to the viewer (if his head is actually in that average position) to be equally spaced across a horizontal line on the display screen 500, and the minimum movement through which the head of an avatar will be seen to turn to look from one avatar to another or from one avatar to the user is maximised.

A routine for performing the processing described above at steps S500 to S512 is given in Appendix A, in which:

steps S500 to S512 overall are performed by part A;

step S500 is performed by parts B and C;

step S502 is performed by part D, part E1, part F, part J and part K;

step S506 is performed by parts E2 and E3, part H, part I and part L; and

steps S508 and S510 are performed by part E4.

Figure 37 shows examples of results of performing steps S500 to S512 for a distance of the viewer from the viewing screen 500 corresponding to three screen half-widths (which has been found in practice to be a typical distance for the viewer when viewing a conventional PC monitor 34).

Referring to Figure 37, the position of the viewer is labelled P1, the positions in the 3D computer model of the conference room of the two outermost avatars to be displayed to the user (irrespective of the total number of avatars to be displayed) are shown by the solid circles 800 and 810, while the position in the 3D model of the remaining avatar when three avatars are to be displayed is shown by diamond 820, the positions of the remaining two avatars when four avatars to be displayed are shown by squares 830 and 840, the positions of the remaining three avatars in the 3D model when five avatars are to be displayed are shown by triangles 850, 860 and 870, and the positions of the remaining six avatars in the 3D model when eight avatars are to be displayed are shown by circles 880, 890, 900, 910, 920 and 930. All coordinate values shown in Figure 37 are expressed as multiples of the screen half-width.

Referring again to Figure 33, at step S514, the currently stored avatar configuration is selected as the avatar configuration to be used for the video conference.

When performing processing as set out above to calculate the positions of the avatars in the 3D conference room model, step S26 in Figure 5 (at which the conference coordinator selects the shape of the conference room table) is unnecessary. In addition, when performing step S68 in Figure 7, rather than selecting a model of the conference room table in accordance with the instructions from the conference coordinator and then determining the positions of the avatars around the table, the positions of the avatars are determined as described above and then a 3D conference room table model is defined to fit between them.

In the processing described above, the average distance of the viewer from the display screen 500 is calculated at step S522 and S526 (Figure 34), and the calculated average distance is used to compute positions in the 3D computer model of the conference room for the avatars to be displayed to the user. In this way, the computed configuration remains fixed throughout the video conference. However, instead, the actual distance of the user from the display screen 500 may be monitored during the video conference, and the positions of the avatars in the 3D computer model changed as the distance of the user from the display screen changes. In this case, rather than perform the processing described above to re-calculate the positions of the avatars in the 3D computer model of the conference room for each new distance of the user from the display screen, the calculations may be performed beforehand and stored in a look-up-table which defines the positions of the avatars in the 3D computer model of the conference room for different distances of the user from the display screen 500. The look-up-table may also define these positions for a different number of avatars to be displayed, thereby enabling it to be used for different video conferences.

Figure 38 shows an example of a look-up-table 1000, in which the positions of the avatars in a 3D conference room model are defined for distances of the user from the display screen 500 corresponding to twice the screen half-width, three times the screen half-width, four times the screen half-width, and five times the screen half-width (although in practice avatar positions would be defined for many other values of the distance of the user from the display screen 500). In addition, the positions are defined for three, four, five, six, seven and eight avatars to be displayed.

During the video conference, the actual position of the user from the display screen 500 may be calculated and used as an input to the look-up-table to read the positions in the 3D computer model of the conference room for the distance of the user in the look-up-table which is closest to the calculated actual distance of the user.

A look-up-table may also be stored and used to determine the positions of the avatars in the 3D conference room model even when the positions are to remain fixed throughout the video conference. For example, the average distance of the user from the display screen may be used as an input to the look-up-table to read the positions in the 3D computer model of the conference room for the distance of the user in the look-up-table which is closest to the input average distance of the user.

Appendix A

```
A     FindOptimumLayout
A1    B SetUp()                                   // This reads in input values
A2    R = 0                                       // Initialise the recursion counter
A3    D DoRecursion()
A4    E ComputeConfig(x*)
A5    END

B0    SetUp()
B1    Number of Virtual Participants = V (input)  // Number of displayed avatars in screen
B2    N=V+1
B3    Degrees of freedom D = floor( (V-1) /2 )
B4    Number of non symmetrically redundant VPs U = ceil( V/2 )
B5    MinScale = 0.3 (input)                      // The minimum scale acceptable relative to
                                                  // the peripheral avatars
B6    MaxScale = 1.0 (input)                      // The maximum scale acceptable relative to
                                                  // the peripheral avatars
B7    Step = 0.01 (input)                         // The search step size or resolution
B8    parallax = TRUE (input)                     // A flag to indicate that the turn should be
                                                  // scaled by the depth (x-coordinate) of the
                                                  // avatar, to generate apparent turn or parallax
B9    K = distance between the viewer and the display (as a multiple of screen half-width) (input)
B10   Xmax = C XfromScale(MinScale)
B11   Xmin = C XfromScale(MaxScale)
B12   OptCritTurn = -1                            // Set to an arbitrary small number.
B13   x is a D-vector (a vector with D elements, one for each degree of freedom) x[0..D-1]

C     XfromScale(s) – converts scale to depth,x.
C1    Return X = K*(1/s –1)

D     DoRecursion()
D1    If (R<D)
      {
D2          R++
D3          If (R==1)
D4                For (x[0]=xMin; x[0]<xMax; x[0]+=Step)   // For x[0] start search from
                        D DoRecursion();                   // Xmin
            Else
D5                For (x[R-1]=x[R-2]; x[R-1]<xMax; x[R-1]+=Step)   // For x[i] i>1,
                        D DoRecursion();                   // start search from x[i-1]
D6          R--
      }
      else
      {
D7          E ComputConfig(x)
      }
      RETURN

E     ComputeConfig(x)
E1    F ComputeVPPositions(x) // Compute the positions of the Virtual Participants or avatars
E2    H ComputeTurnAngles(x) // Now compute the angles or parallax
E3    I FindCriticalTurn();       // Find the smallest (critical) turn/parallax for this configuration
E4    If (critTurn>optCritTurn) x* = x, optCritTurn = critTurn     // x* is the optimum
                                                                  // configuration
```

```
E5      RETURN

F       ComputeVPPositions(x)
F1      p[0].SetX(-m_k); p[0].SetY(0.0); // ...The position of the Real Participant
F2      p[1].SetX(0.0);   p[1].SetY(1.0); // ...The upper-most avatar
F3      for (u = 2; u<=U; u++)
        {
F4        xu = x[u-2]                          // Upper avatar
F5        yu = J Y(u,xu);
F6        p[u].SetX(xu), p[u].SetY(yu)
F7        If (u!=V-u+1)
          {
F8          p[V-u+1].SetX(xu), p[V-u+1].SetY(yu)  // Lower avatar
          }
F9      p[v].SetX(0.0),p[v].SetY(-1.0) // ...The lower avatar
        RETURN

H       ComputeTurnAngles(x)
H1      for (j=1; j<=U; j++) {
H2              for (i=0;i<=V;i++) {  // Angle between j looking at i and i+1...
H3                      if (j==i || j==((i+1)%N))
H4                              turn(j-1,i) = 999          // an arbitrarily large number
                        else
                        {
H5                              vji = p[i] - p[j]
H6                              vji.Normalise()
H7                              vjip1 = p[(i+1)%N] - p[j]
H8                              vjip1.Normalise()
H9                              tau = ArcCos(vji.DotProduct(vjip1)
H10                             If (parallax)  // scale the angle if apparent turn is required
H11                                     Tau *= L ScaleFromX (p[i].X())
H12                             turn(j-1,i) = tau
                        }
                }
        }
        RETURN

I       FindCriticalTurn();
I1      CritTurn=999.0   // Set to an arbitrary large number
I2      For (j=1; j<=U; j++)
I3              For (i=0;i<=V;i++)   // Angle between j looking at i and i+1...
I4                      if (turn(j-1,i)<critTurn) {
I5                              CritTurn = turn(j-1,i)
        RETURN critTurn

J       Y(u,xx)
J1      RETURN K ProjPosn(u)*( K + xx ) / K

K       ProjPosn(u)              // Caluclates the y-coordinate of the projection of the u^th avatar
K1      gap = 2/(V-1)            // gap is the distance between the projections of the avatars
K2      RETURN 1.0 – (u-1)*gap

L       ScaleFromX(xx)
L1      Return K/(K+xx);
```

CLAIMS

1. A computer conferencing system operable to carry out a conference by animating three-dimensional computer models of the participants in dependence upon real-world movements thereof, the system comprising a plurality of user stations arranged to generate and exchange data so that each user station displays a sequence of images of a respective three-dimensional computer model containing three-dimensional computer models of the participants at the other user stations, and such that movements of at least the heads of the participants in real-life produce corresponding movements of the three-dimensional computer models, wherein each user station comprises:

storage means storing data defining a three-dimensional conference computer model containing a three-dimensional computer model of each participant at the other user stations, the three-dimensional conference computer model being different from the three-dimensional conference computer model stored at each of the other user stations in the system;

means for generating and displaying images of the three-dimensional conference computer model, wherein the content of the displayed images is independent of the movement of the head of each viewing participant;

means for determining and outputting the position in the displayed images at which each participant at the user station is looking; and

processing means for moving at least the head of the three-dimensional computer model of each participant in dependence upon data received from the other participants, so that images displayed at the user station convey the head movements of the participants.

2. A system according to claim 1, wherein:

each user station further comprises means for recording and outputting image data of at least the head of each participant at the user station; and

each user station is arranged to generate the image data for display by rendering the image data for a participant onto the corresponding three-dimensional computer model.

3. A system according to claim 1 or claim 2, wherein the three-dimensional conference computer model at each user station further contains a three-dimensional computer model of a character to be animated during the conference but whose head movements during the animation are not determined by the head movements of a participant to the conference.

4. Computer processing apparatus for use in a computer conferencing system according to claim 1, comprising:

means for storing data defining a three-dimensional conference computer model containing a three-dimensional computer model of each participant at the other apparatus in the system;

means for generating image data for the display of images of the three-dimensional conference computer model, such that the content of the displayed images will be independent of movements of the heads of viewing participants;

means for determining and outputting the position in a displayed image at which each participant at the user station is looking; and

processing means for moving at least the head of the three-dimensional computer model of each participant in dependence upon data received from the other participants so that the displayed images will convey the head

movements of the participants.

5. Apparatus according to claim 4, wherein:

the apparatus further comprises means for outputting image data of the head of each participant at the apparatus; and

the apparatus is arranged to generate the image data for display by rendering received image data for a participant onto the corresponding three-dimensional computer model.

6. Apparatus according to claim 4 or claim 5, further comprising means for generating data defining the three-dimensional conference computer model in accordance with a seating plan of the participants.

7. Apparatus according to claim 6, wherein:

the means for generating the data defining the three-dimensional conference computer model is arranged to determine positions in the three-dimensional conference computer model for the three-dimensional computer models of the participants in dependence upon the seating plan, the width of the display on which the images of the three-dimensional conference computer model are to be displayed and a distance of a viewing participant from the display; and

the means for generating the image data for display is arranged to generate the image data by rendering the three-dimensional conference computer model from a position defined by the distance of the viewing participant from the display used to generate the data defining the three-dimensional conference computer model.

8. Apparatus according to claim 7, wherein the means for generating the data defining the three-dimensional conference computer model and the means for generating the image data for display are arranged to change the positions in the three-dimensional conference computer model of at least one of the three-dimensional computer models of the participants and the position from which the three-dimensional conference computer model is rendered to generate the image data for display as the distance of the viewing participant from the display changes during the conference.

9. Apparatus according to claim 7 or claim 8, wherein the means for generating the data defining the three-dimensional conference computer model is arranged to determine the positions for the three-dimensional computer models of the participants such that, in the images displayed to the viewing participant, the three-dimensional computer models of the participants are substantially equally spaced across the display and the minimum movement which the head of a three-dimensional computer model of a participant undergoes to look from one participant to another is maximised.

10. Apparatus according to any of claims 4 to 9, further comprising means for generating and outputting data defining movements of at least one body part other than the head of each viewing participant.

11. Apparatus according to claim 10, wherein the means for generating the data defining movements is arranged to generate data defining the three-dimensional positions of discrete points on each viewing participant.

12. Apparatus according to claim 10 or claim 11, wherein the means for generating the data defining movements comprises means for processing signals defining images of each viewing participant to generate the data defining the movements.

13. Apparatus according to claim 12, wherein the means for generating the data defining movements comprises means for processing image data from a plurality of cameras to generate the data defining the movements.

14. Apparatus according to claim 13, wherein the means for generating the data defining movements comprises means for matching feature points in images from respective cameras to generate the data defining the movements.

15. Apparatus according to claim 14, wherein the feature points comprise at least one of lights and coloured markers.

16. Apparatus according to any of claims 4 to 15, wherein the means for determining the position in a displayed image at which a participant is looking comprises means for generating data defining the position relative to the participants displayed in the image.

17. Apparatus according to any of claims 4 to 16, wherein the means for determining the position in a displayed image at which a participant is looking comprises means for processing signals defining images of the participant to generate the data defining the position.

18. Apparatus according to claim 17, wherein the means for determining the position in a displayed image at which a participant is looking is arranged to determine the position in dependence upon the position of the participant's head.

19. Apparatus according to claim 18, wherein the means for determining the position in a displayed image at which a participant is looking is arranged to determine the position by determining a plane representing the position of the participant's head and projecting a line from the plane to the displayed image.

20. Apparatus according to any of claims 4 to 19, further comprising calibration means for performing calibration processing to determine the position of a display screen on which the image data will be displayed.

21. Apparatus according to claim 20, wherein the calibration means is arranged to determine the position of the display screen by determining the plane of the display screen and determining the position of the plane in three dimensions.

22. Apparatus according to claim 21, wherein the calibration means is arranged to determine the plane of the display screen and the position of the plane in dependence upon the configuration of the participant's head when looking at known positions on the display screen.

23. Apparatus according to any of claims 4 to 22, further comprising

display means for displaying the image data.

24. Apparatus for connection to a plurality of corresponding apparatus to carry out a virtual meeting by animating participant avatars in dependence upon movements of the real participants, wherein the apparatus is arranged to store and animate a 3D computer model of the meeting which is different to the 3D computer model stored at the corresponding apparatus.

25. A method of carrying out a computer conference by animating three-dimensional computer models of the participants in dependence upon real-world movements thereof, wherein data is exchanged between a plurality of user stations so that each user station displays a sequence of images of a respective three-dimensional computer model containing three-dimensional computer models of the participants at the other user stations, and such that at least head movements of the participants in real-life produce corresponding movements of the three-dimensional computer models, wherein each user station is operated such that:

data is stored defining a three-dimensional conference computer model containing a three-dimensional computer model of each participant at the other user stations, the three-dimensional conference computer model being different from the three-dimensional conference computer models stored at each of the other user stations in the system;

images are generated and displayed of the three-dimensional conference computer model, wherein the content of the displayed images is independent of the movement of the head of each viewing participant;

the position in the displayed images at which each participant at the

user station is looking is determined and output; and

at least the heads of the three-dimensional computer models of the participants are moved in dependence upon data received from the other participants, so that images displayed at the user station convey the movements of the participants.

26. A method according to claim 25, wherein, at each user station:

image data of at least the head of each participant at the user station is recorded and output; and

the image data for display is generated by rendering the image data for a participant onto the corresponding three-dimensional computer model.

27. A method according to claim 25 or claim 26, wherein the three-dimensional conference computer model at each user station further contains a three-dimensional computer model of a character to be animated during the conference but whose head movements during the animation are not determined by the head movements of a participant to the conference.

28. A method of operating a computer processing apparatus in a computer conferencing system to carry out a conference between participants at a plurality of apparatus, comprising:

storing data defining a three-dimensional conference computer model containing a three-dimensional computer model of each participant at the other apparatus;

generating image data for the display of images of the three-dimensional conference computer model, such that the content of the displayed

images will be independent of movements of the heads of viewing participants;

determining and outputting the position in a displayed image at which each participant at the user station is looking; and

moving at least the heads of the three-dimensional computer models of the participants in dependence upon data received from the other participants so that the displayed images will convey the head movements of the participants.

29. A method according to claim 28, wherein:

the method further comprises recording and outputting image data of the head of each participant at the apparatus; and

the image data for display is generated by rendering received image data for a participant onto the corresponding three-dimensional computer model.

30. A method according to claim 28 or claim 29, further comprising the step of generating data defining the three-dimensional conference computer model in accordance with a seating plan of the participants.

31. A method according to claim 30, wherein:

in the step of generating the data defining the three-dimensional conference computer model, positions in the three-dimensional conference computer model for the three-dimensional computer models of the participants are determined in dependence upon the seating plan, the width of the display on which the images of the three-dimensional conference computer model are to be displayed and a distance of a viewing participant from the display; and

in the step of generating the image data for display, the image data is generated by rendering the three-dimensional conference computer model from a position defined by the distance of the viewing participant from the display used to generate the data defining the three-dimensional conference computer model.

32. A method according to claim 31, wherein, in the steps of generating the data defining the three-dimensional conference computer model and generating the image data for display, the positions in the three-dimensional conference computer model of at least one of the three-dimensional computer models of the participants and the position from which the three-dimensional conference computer model is rendered to generate the image data for display are changed as the distance of the viewing participant from the display changes during the conference.

33. A method according to claim 31 or claim 32, wherein, in the step of generating the data defining the three-dimensional conference computer model, the positions for the three-dimensional computer models of the participants are determined such that, in the images displayed to the viewing participant, the three-dimensional computer models of the participants are substantially equally spaced across the display and the minimum movement which the head of a three-dimensional computer model of a participant undergoes to look from one participant to another is maximised.

34. A method according to any of claims 28 to 33, further comprising the step of generating and outputting data defining movements of at least one body

part other than the head of each viewing participant.

35. A method according to claim 34, wherein the data defining movements is data defining the three-dimensional positions of discrete points.

36. A method according to claim 34 or claim 35, wherein the data defining movements is generated by processing signals defining images.

37. A method according to claim 36, wherein the data defining movements is generated by processing image data from a plurality of cameras.

38. A method according to claim 37, wherein the data defining movements is generated by matching feature points in images from respective cameras.

39. A method according to claim 38, wherein the feature points comprise at least one of lights and coloured markers.

40. A method according to any of claims 28 to 39, wherein the data defining the position in a displayed image at which a participant is looking comprises data defining the position relative to the participants displayed in the image.

41. A method according to any of claims 28 to 40, wherein the position in a displayed image at which a participant is looking is generated by processing signals defining images of the participant to generate the data defining the position.

42. A method according to claim 41, wherein the position in a displayed image at which a participant is looking is determined in dependence upon the position of the participant's head.

43. A method according to claim 42, wherein the position in a displayed image at which a participant is looking is determined by determining a plane representing the position of the participant's head and projecting a line from the plane to the displayed image.

44. A method according to any of claims 28 to 43, further comprising a calibration step of performing processing to determine the position of a display screen on which the image data will be displayed.

45. A method according to claim 44, wherein the calibration step determines the position of the display screen by determining the plane of the display screen and determining the position of the plane in three dimensions.

46. A method according to claim 45, wherein the calibration step determines the plane of the display screen and the position of the plane in dependence upon the configuration of the participant's head when looking at known positions on the display screen.

47. A method according to any of claims 28 to 46, further comprising a step of displaying the image data.

48. A method of operating a computer processing apparatus to carry out a

virtual meeting by animating participant avatars in dependence upon movements of the real participants, wherein movements of at least the participants' heads produce corresponding movements of the avatars in a three-dimensional computer model of the conference which is different to the three-dimensional computer model of the conference at each other computer processing apparatus participating in the conference.

49. A storage medium storing computer-useable instructions, which, when loaded into a programmable computer processing apparatus, enable the apparatus to become configured as an apparatus according to at least one of claims 4 to 24.

50. A signal conveying computer-useable instructions, which, when loaded into a programmable computer processing apparatus, enable the apparatus to become configured as an apparatus according to at least one of claims 4 to 24.

51. A storage medium storing computer-useable instructions, which, when loaded into a programmable computer processing apparatus, enable the apparatus to become operable to perform a method according to at least one of claims 28 to 48.

52. A signal conveying computer-useable instructions, which, when loaded into a programmable computer processing apparatus, enable the apparatus to become operable to perform a method according to at least one of claims 28 to 48.

PATENTS • DESIGNS • COPYRIGHT • TRADE MARKS
The Patent Office
95

INVESTOR IN PEOPLE

Application No: GB 0000088.5
Claims searched: ALL
Examiner: R. F. King
Date of search: 13 October 2000

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): H4T[TBAS, TBBA, TBBD, TCGD, TBEX, TCJA, TCXX]

Int Cl (Ed.7): G06T13/00, 15/70, 17/00

Other: ONLINE: EPOQUE.

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
A	EP0696018 A2	[Nippon Telegraph] See abstract	1
"	WO00/10099 A1	[NET TALK] See abstract	"
"	WO99/30494 A1	[Brit. Telecom] See abstract	"
"	US5,491,743 A	[IBM] See abstract	"

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

THIS PAGE BLANK (USPTO)

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☑ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.